AUTOMATED WAVELET ANALYSIS OF LOW RESOLUTION GAMMA-RAY SPECTRA AND PEAK AREA UNCERTAINTY

BY

HAO XIONG

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Nuclear, Plasma and Radiological Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2015

Urbana, Illinois

Master's Committee:

Assistant Professor Clair J. Sullivan, Chair
Associate Professor Ling Jian Meng

ABSTRACT

The accuracy of automated isotope identification from low resolution gamma-ray spectra can be significantly improved with better algorithms. The method based on the wavelet transform and non-negative least squares (NNLS) are discussed in this thesis. Several improvements are made for the wavelet algorithm itself and different options can be configured in the MATLAB code. The partial or whole spectrum can be sent to NNLS and analyzed with or without subtracting the continuum. The boundary effects are also discussed. Several methods are developed to determine the area uncertainty. The matrix form of wavelet transform and error propagation are used. The inversion of the basis matrix is obtained either by the Moore-Penrose pseudo inversion or by truncated singular value decomposition (TSVD). The results are compared with those given by OriginLab and Gaussian fitting in MATLAB, which are consistent with each other, while TSVD is shown to be more accurate. The wavelet algorithm using TSVD for the area uncertainty calculation works well for complicated spectrum continuum and for overlapping peaks.

To my parents, for their love and support.

ACKNOWLEDGMENTS

First and foremost, I would like to thank my parents and family for their endless love and support on my road to knowledge.

I would like to thank Dr. Clair Sullivan for all the patient and kind help throughout the entire research. I am very lucky to do research in Dr. Sullivan's group and I really learned a lot here.

I am very grateful for the help of Dr. Meng Lingjian and Dr. James Stubbins and thank them for their efforts on improving this thesis.

I would also thank my colleagues Jacob Stinnett, Mark Kamuda and Yi Liu and all other members in Dr. Sullivan's group for helpful discussions.

Finally, I sincerely thank Becky Meline, Gail Krueger and all other faculty and staff of NPRE for their help and guidance.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CWT	Continuous Wavelet Transform
DRF	Detector Response Function
NNLS	Non-negative Least Square
B	Basis Matrix
B1	Basis Matrix used in NNLS
S	Coefficient Vector
C_S	Covariance Matrix of Coefficient Vector
B1	Base Matrix with Several Columns Selected
k	Fitted Vector of NNLS
WTMM	Wavelet Transform Modulus Maxima
PI	Pseudo Inversion
TSVD	Truncated Singular Value Decomposition
σ	Standard Deviation
σ_i	Standard Deviation of the i-th Peak
σ_n	Standard Deviation of Noise
σ_x	Standard Deviation of Peak
x	Spectrum
Cx	Covariance Matrix of Spectrum
μ	Peak Centroid
μ_i	Peak Centroid of the i-th One

A	Peak Area
σ_A	Peak Area Uncertainty
N	Number of Peaks

CHAPTER 1

INTRODUCTION

Low resolution gamma-ray spectrometers such as sodium iodide (NaI) are widely used for automated isotope identification, but their accuracy requires significant improvement through improved algorithms [1–3]. Several methods such as library comparisons, region of interest (ROI), template matching, expert interaction and so on have been developed, however they provide inaccurate results due to calibration shift, shielding, overlapping peaks and peak areas [4]. The accuracy of isotope identification from low resolution gamma-ray spectra can be significantly improved by better designed algorithms. A method based on Bayesian statistics was recently published, which took the centroid and area of the peaks as input and returned the posterior probability of the presence of each isotope in the isotope library [5–7].

The difficulty to determine the peak centroid and area comes from the unresolved photopeaks and background noise, especially when the counting time is short. The overlapped peaks can be fit through multiple Gaussian functions, which can be done with OriginLab or MATLAB [8]. This method requires knowledge of the number of the overlapping peaks and may be sensitive to high frequency noise. A newly developed method can overcome such problems by using the continuous wavelet transform (CWT) and nonnegative least squares (NNLS) fitting [9–13].

1.1 Isotope Identification Problem

An isotope is characterized by its gamma-ray spectrum and many isotope identification methods usually require a reliable way to obtain the peak centroids and areas from the spectrum. Photopeaks in gamma spectra are typ-

ically Gaussian functions added to the Compton continuum and statistical Poissonian noise. The Compton continuum is generated because of the leakage of energy in the form of the scattered photons. Many isotopes consist of high energy photopeaks but lower energy photopeaks may reside above the Compton continuum of ones with higher energy.

Because of the above complexities, it is difficult to obtain the peak area. One natural method is Gaussian fitting with the continuum subtracted. On one hand, the continuum may not be linear and is difficult to estimate. On the other hand, it is not easy to fit the spectrum if multiple photopeaks overlap. For a single photopeak with good statistics, the area and its uncertainty can be obtained using the Covell method [14]. This method locates the highest channel in the peak and mark the peak limits an equal number of channels away from the centroid channel. The limits can be extended down the sides of the peak to the background continuum level which is called the total peak area method.

1.2 Motivation

The wavelet transform is powerful in spectral analysis. So it may be useful to analyze gamma spectrum in the presence of counting noise. The photopeaks are Gaussian functions and this may be used to resolve the overlapping problems. A new method combining all these ideas was developed recently and showed promising applications [9–12].

In this method, the continuous wavelet transform (CWT) is applied to the spectrum, which produces a 2D coefficient matrix called scalogram. A non-negative least squares fitting is used with a specially constructed basis matrix to find the centroids and areas of the overlapping peaks.

With the CWT, the spectrum can be analyzed in the scalogram, which is actually a multi-resolution analysis. The scalogram depends on the channels numbers and scales used in the transform. Low scale indicates high frequency and high scale indicates low frequency. Counting noise is evenly distributed at all scales while the signal is localized at a certain scale. By choosing the

proper scale, this technique may help to separate the noise from useful spectral feature to a certain degree.

The columns of the basis matrix is the CWT of Gaussian functions with unit area and with centroids at different channels. This acts like a moving selector which is capable of identifying Gaussian functions in the spectrum.

1.3 Difficulty

The wavelet algorithm introduced in the previous section is promising in isotope identification. Before its full application, two main problems need to be solved: (1) The boundary effect problem and (2) The calculation of photopeak area uncertainty: the uncertainty of the calculated area between the Gaussian curve and the continuum. These are the main tasks to be solved in this thesis.

The boundary effect problem comes from the fact that the spectrum has finite length. If a certain region is extracted from the spectrum for analysis, there will be a sharp jump at the edge. This artificial boundary can degrade the performance of the wavelet algorithm.

Additionally, the continuum may vary greatly and lead to false peaks given by the wavelet algorithm. To solve this problem, the calculation of the peak area uncertainty is necessary which originates from the error propagation. Here the error originates with the Poissonian noise. This error propagation is difficult to analyze because inversion of singular matrix is needed in our method.

1.4 Thesis Structure

It has been demonstrated that the wavelet method can give information of the peak centroid and area of overlapped peaks, while its full application requires the calculation of the area uncertainty as well, including further improvements of the wavelet algorithm itself [12]. Possible improvements re-

lated to improving the insensitivity to the Compton continuum and Poissonian noise are discussed in detail in Chapter 2; Examples of their application in simulation and experiments are explained in Chapter 3 and 4. The calculation of peak area uncertainty and its applications are discussed throughout these chapters as well. The effectiveness of the improved wavelet algorithm can be seen in Chapter 4, which shows its successful application to spectra with complicated continuum and overlapping peaks.

The eventual goal of the wavelet algorithm is to correctly find the peak centroids, areas and uncertainties in areas and provide that as input to the Bayesian algorithm. To date, this has been achieved for a variety of simulated and measured spectra. More experimental tests are needed for parameter optimization, which will be the main research work in the future.

CHAPTER 2

THEORY

The wavelet algorithm developed in this research is capable of resolving overlapping photopeaks in gamma spectra based on the application of the detector response function (DRF), continuous wavelet transform (CWT), non-negative least squares (NNLS) fitting and error propagation estimation. The main idea of using wavelet analysis in isotope identification (ID) has been explored for years, which consists of the following parts:

1. Perform the CWT of the gamma spectrum and get a 2D coefficient matrix (called the scalogram) depending on scale and channel number;

2. Link all the local maximum points in the scalogram and get a variety of lines called Wavelet Transform Modulus Maxima (WTMM) Lines [15];

3. Find the maximum point along each WTMM line, apply some filtering criterion and get the points (S_{max}) corresponding to true photopeaks;

4. For each local maxima found in the previous step, perform NNLS fitting using a measured basis matrix. The fit results give the peak centroids and areas.

The above steps will be explained in detail in the following sections. In this research, some improvements have been made:

1. Improvement of the filtering criterion used in step 3, which can extract more true photopeaks;

2. Improvement of the NNLS fitting and possible solutions to the boundary effect problems;

3. A variety of methods of area uncertainty calculation.

The main focus will be put on the boundary effect problems and area uncertainties throughout the thesis, which are the main difficulties in the wavelet algorithm.

After an introduction to the wavelet transform, we will then discuss the the application of wavelet analysis in our research [16–18]. The detailed process of non-negative least squares and its related problems will be elaborated as well as an in-depth comparison of all possible techniques of the peak area estimation [19–21].

2.1 Wavelet Transform

A gamma spectrum is typically a series of data which shows the counts or counting rates at different channel numbers or energies in a corresponding waveform. If the spectrum satisfies certain conditions (e.g. absolutely integrable or square integrable), it can be decomposed to a series of cosine and sine functions with different frequencies mathematically, which is called Fourier series. For a spectrum with infinite length ($L \to \infty$), this is generalized to the so-called Fourier transform.

$$x(t) = \int_{-\infty}^{\infty} X(\omega) e^{i\omega t} dt \tag{2.1}$$

$$X(\omega) = \int_{-\infty}^{\infty} x(t) e^{-i\omega t} dt \tag{2.2}$$

Through the Fourier transform, the components at different frequencies in a spectrum can be seen clearly in the transformed form. However, the spectrum can only be analyzed in the time or frequency domain separately. It could be possible that two different spectra produce the same Fourier transform, as shown in Figure 2.1.

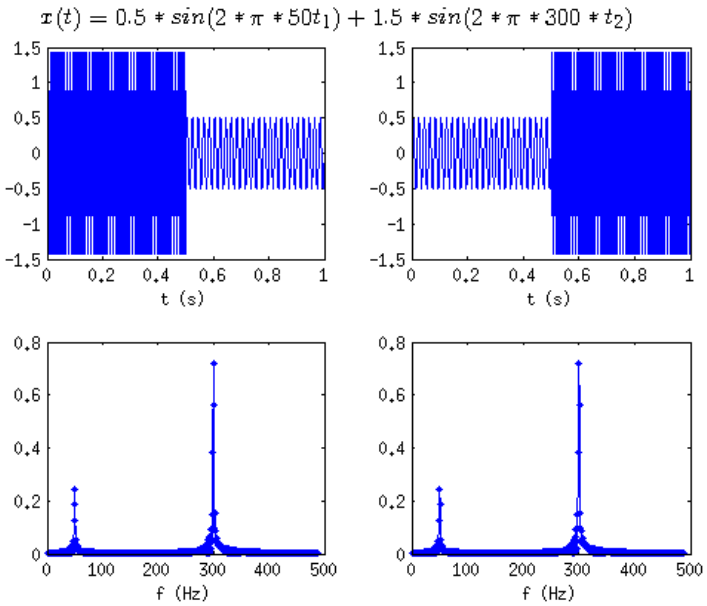


Figure 2.1: Two different spectra (top) with the same Fourier transform (bottom).

One solution to resolve this is to use the Short Time Fourier Transform (STFT), which divides the time-series into segments and performs Fourier transform in each segment.

$$X(\tau, \omega) = \int_{-\infty}^{\infty} x(t) w(t - \tau) e^{-i\omega t} dt \tag{2.3}$$

where $w(t)$ is a window function (e.g. square function, Gaussian function et. al.). This can be viewed as a convolution of the original spectrum and the window function (fixed length). There are many choices for the window function with different applications, as shown in Figure 2.2.

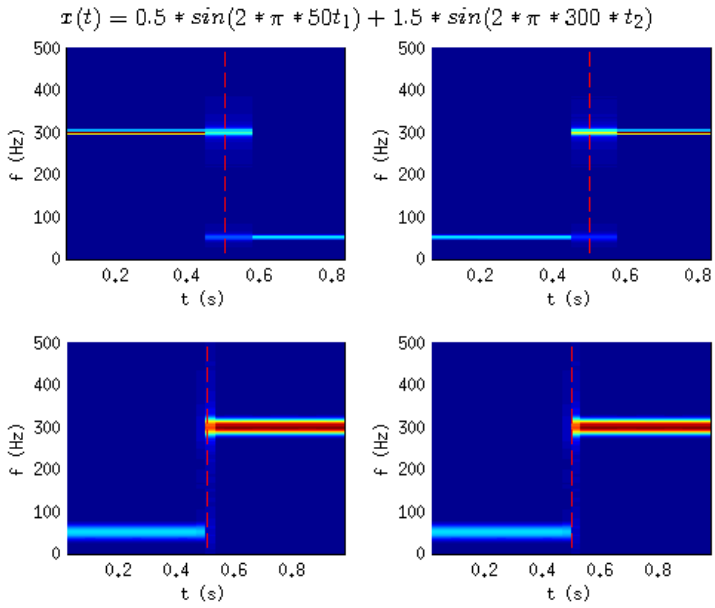


Figure 2.2: Example of STFT with different window length.

From the above figure, it can be seen that the time and frequency resolution cannot be simultaneously good enough at the same time.

The continuous wavelet transform (CWT) takes a further step by using window functions with variable length.

$$T(E,S) = \int_{-\infty}^{\infty} f(t)\frac{1}{\sqrt{S}}\psi^*(\frac{t-E}{S})dt \tag{2.4}$$

where t is the independent variable, $f(t)$ is the spectrum, ψ is the mother wavelet function, E, S are two parameters that shift and scale the mother wavelet function and $T(E,S)$ is the 2D coefficient matrix produced by the CWT.

The produced result of the wavelet transform is a 2D coefficient matrix which depends on time and scale (or frequency). It can be seen clearly from Figure 2.3 that the frequency resolution varies at different frequencies. Because of this, wavelet analysis is often called multi-resolution analysis.

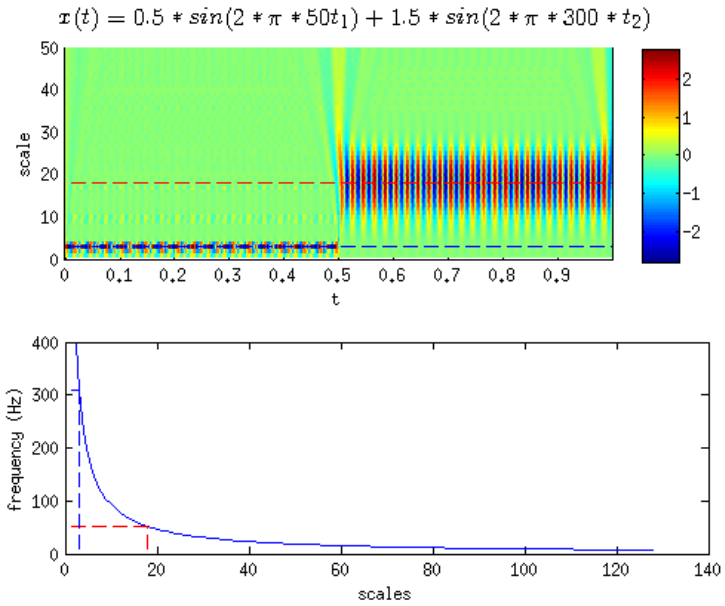


Figure 2.3: Example of wavelet transform of the same signal as above.

A wavelet function should satisfy

$$\int_{-\infty}^{\infty} \psi(t)dt = 0. \tag{2.5}$$

All functions with zero integral and finite duration are wavelets. It can be easily seen that a wavelet transform of a constant is zero, which means horizontal baseline will be removed automatically. We can see later that linear baseline will be removed as well for some wavelet functions if the integral is taken from $-\infty$ to ∞. Note that in practice, the wavelet transform is taken for a signal with finite length in reality. This could lead to problems related to boundary effects which can be possibly solved in several ways. We will discuss about this in detail in the following chapters.

The wavelet transform produces a coefficient matrix (scalogram) that is closely related to the matching between the signal and wavelet. By choosing a proper wavelet function, we can detect shape changes in the spectrum from the scalogram. A biorthogonal wavelet function ('bior2.6' in MATLAB, see Figure 2.4) will be used throughout this research as it exhibits best performance [12, 22]. Additionally, the wavelet transform can be used in band pass filtering, denoising and structure matching as well.

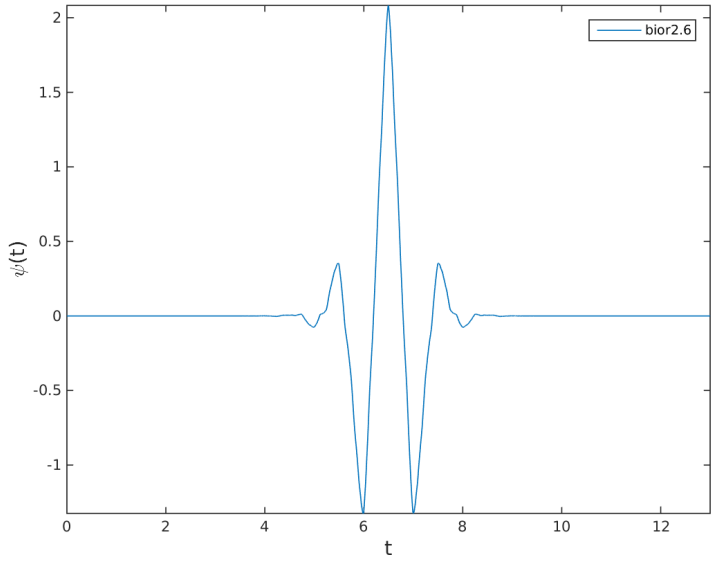


Figure 2.4: bior2.6 wavelet function.

2.2 Wavelet Analysis at Optimal Scale

In this thesis, we used a new algorithm to find all WTMM lines in the scalogram. This process can be summarized as the following pseudo code:

```
define the minimum distance h of two points in different lines
define a matrix as the same size as the coefficient matrix and initialize to zero (flag matrix)
for scale range from 1 to maximum
  get a list of channels of the maximum values in each row of the coefficient matrix (list L)
  for point p in list L
    add the values of the sub matrix (2h columns and 2h rows with point p in the center) around point p in the flag matrix (sum)
    if sum is zero
      create a new WTMM line and add the point to that line
      set the point in the flag matrix to the value of the line index
    else
      determine the index of the line that the point belongs to
      add the new found point to the corresponding existed line
    end
  end
end
```

Peaks in gamma spectra can be modeled as Gaussian functions with the form:

$$f = \frac{A}{\sqrt{2\pi}\sigma} exp(-\frac{(x-\mu)^2}{2\sigma^2}) \tag{2.6}$$

where σ is standard deviation, x is channel number μ is peak centroid and A is peak area. For Gaussian distribution, we know that

$$\sigma \propto \sqrt{\mu}. \tag{2.7}$$

For a particular detector, this means the standard deviation of the Gaussian peak should not be too small at a specific channel. If we plot the standard deviation of the Gaussian peak at different channels, we get the Detector Response Function (DRF). There is a linear relationship between the Full Width Half Maximum (FWHM) and the local maxima scale. This can be verified as shown in Figure 2.5:

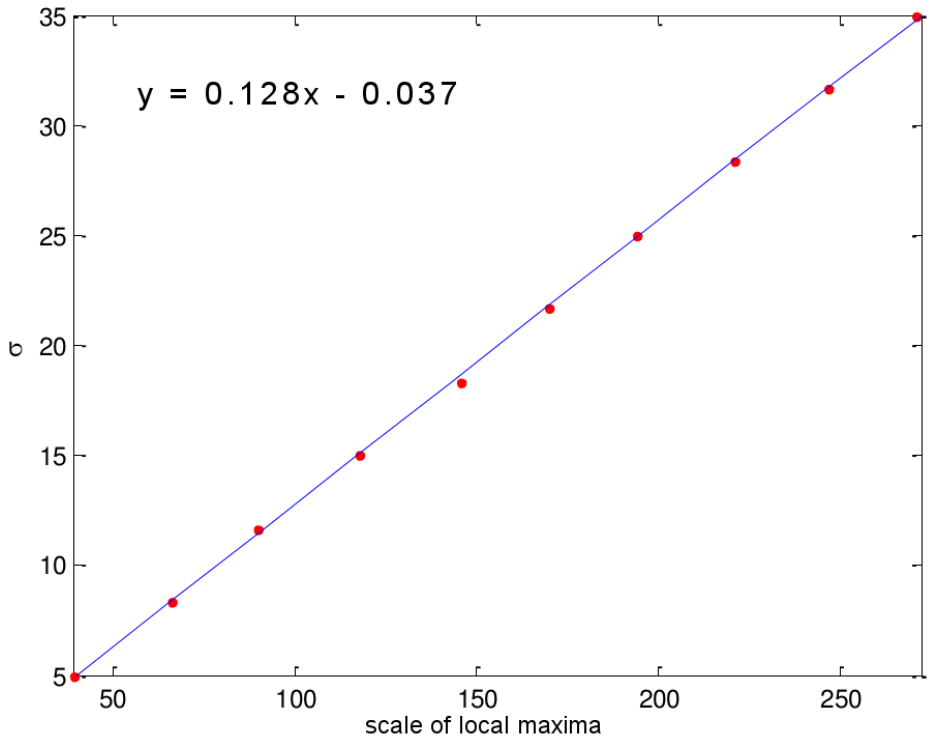


Figure 2.5: Linear relationship between standard deviation of a Gaussian peak and the corresponding local maxima scale

The red points show the simulation result by calculating the corresponding scale of local maxima for Gaussian functions with different standard deviations. The blue line is the fit linear relationship. This property will be used later in the calculation of peak area uncertainty. A local maxima may be produced by the overlap of several photopeaks and the previous shown linear relationship defines a range which contains all possible peak centroids. Thus it is only necessary to check the true peaks within this range.

We could experimentally measure the corresponding scale of the local maxima for photopeaks at different channels and this leads to a fit Detector Response Function (DRF) line in the scalogram. The scale given by this DRF line is called optimal scale S_{opt}. For a single photopeak, only local maxima that satisfy $S_{max} > S_{opt}$ (where S_{max} is the scale of local maxima) can be considered as produced by a true peak (See Figure 2.6). This is not the case when multiple photopeaks overlap, which can result in local maxima with $S_{max} < S_{opt}$ (See Figure 2.7). Several other rules can be used to further filtering false WTMM lines. For example, the line should be approximately

vertical. This step may not filter all true photopeaks, which can be filtered further later with the help of area uncertainty.

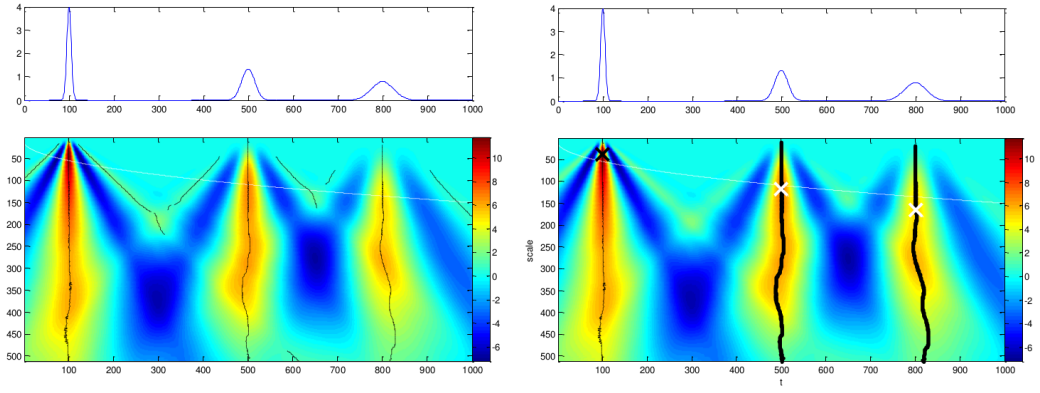
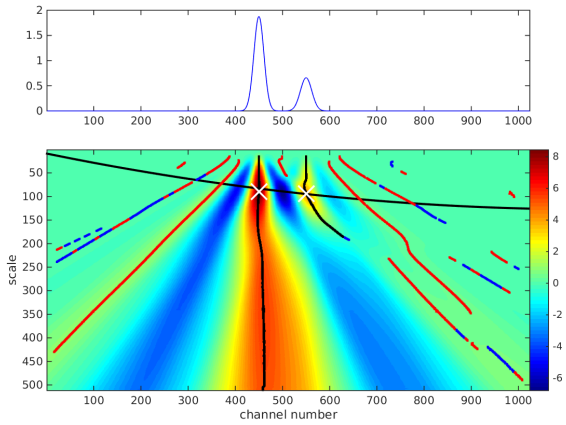


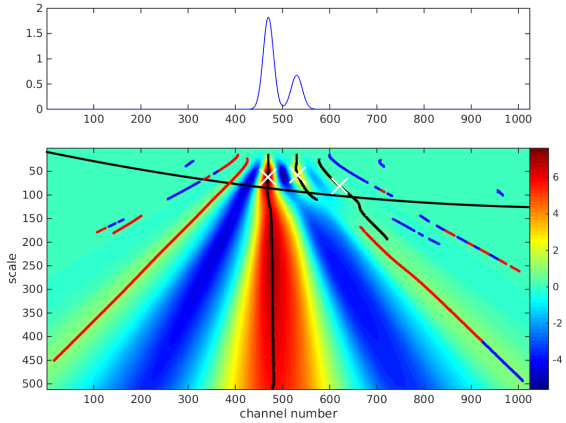
Figure 2.6: Scalogram before and after filtering of the WTMM lines. The black X indicates local maxima filtered out and white X's indicate local maxima produced by true peaks.

There may be more than one local maxima along a WTMM line sometimes. So the first found local maxima may be not always the desired one. One can choose a threshold of scale close to S_{opt}.

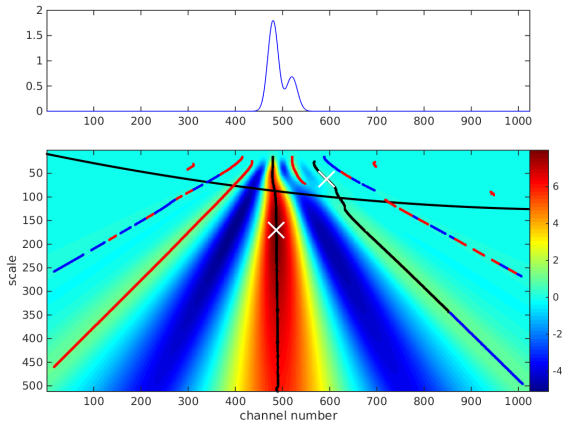
Sometimes a true local maxima may reside slightly above the DRF line. This could happen if the distance between the centroids of the two peaks is approximately 3σ where σ is the standard deviation of the peak (See Figure).



(a) No overlap



(b) Marginal overlap



(c) Significant overlap

Figure 2.7: Different distances between two peaks and the corresponding location of the local maxima.

Thus, the criterion $S_{max} > S_{opt}$ is only valid for a single photopeak while not for overlapping peaks. It is necessary to define a range S that satisfies $S_{max} > S_{opt} - S$ which allows a local maxima to be considered as produced by true photopeaks. This range needs to be optimized by experiments considering the error bar of DRF line as well. The different colors of WTMM lines indicate different lines.

2.3 Non-negative Least Squares Method

Suppose a list of local maxima is returned by the process described in the previous section. Now it is time to extract the peak centroid and area associated with each local maxima. To make it simple at the beginning, we can assume there is only one local maxima in the spectrum. We can simply pad the left and right side of the spectrum with zero to avoid boundary effect.

The NNLS takes the form

$$Bk = S. \tag{2.8}$$

Here B is the basis matrix with size $n \times n$, S is the coefficient vector of wavelet transform at optimal scale with size $n \times 1$. As it is stated in the previous section, the global maxima along a WTMM line may reside far below the DRF line sometimes because of overlap. In this case, it may be reasonable to use the corresponding scale of the maxima in NNLS. The fitted vector k has size $n \times 1$, whose nonzero terms ideally represent the identified peaks (see below). The indexes represent the peak centroids and their values are the corresponding areas. Because of the existence of background noise and the algorithm itself, there are often very small nonzero terms in the vector k which should be filtered out with comparison to the area uncertainty ($k_i < \sigma_i$ with σ_i the area uncertainty of the i-th bin).

The basis matrix is designed as

$$B_{ij} = [CWT(G_j)]_i \tag{2.9}$$

where G_j represents a normalized Gaussian function with centroid j and stan-

dard deviation determined by DRF at this position j; CWT represents the continuous wavelet transform at optimal scale S_{opt} which is a column vector, and $[CWT]_i$ represents its *i-th* element; $S_j = [CWT(x)]_j = a[CWT(G_\mu)]_j$, x represents the original signal, μ represents its centroid and a represents its area. So $Bk = S$ gives

$$\sum_j B_{ij}k_j = S_i \tag{2.10}$$

$$\sum_j [CWT(G_j)]_i k_j = a[CWT(G_\mu)]_j \tag{2.11}$$

$$k_\mu = a \quad (k_j = 0 \text{ if } j \neq \mu). \tag{2.12}$$

For real spectra, $x = x_0 + e$ with e the noise and $S_j = [CWT(x)]_j = a[CWT(G_\mu)]_j + CWT(e)_j$. The resulting k should then be revised by adding k_e which is actually a result of

$$Bk_e = S_e, \quad S_e = CWT(e). \tag{2.13}$$

It is this k_e that accounts for the area uncertainty.

Several factors can lead to non-zero error in the fitted vector k: (1) Background noise; (2) The presence of the Compton continuum; (3) Boundary effects. In the wavelet code, the user can choose whether to use (1) a limited range around the channel number of the local maxima or the whole spectrum; (2) spectrum with fitted baseline subtracted. The boundary effect problem will be discussed in detail in the next chapter. An ideal method is the one that avoids such problems using the original spectrum including radiation background and Compton continuum at a wide channel range. Although the fitted vector k may seem worse (more or larger non-zero error) than those without background noise and at a smaller channel range, it is possible for realistic application with the help of correct peak area uncertainty calculation. At present, this preferred approach worked for the spectra that have been already analyzed. Further statistical analysis with many spectra is needed and will be the main work of this research in the future.

To summarize, the wavelet algorithm will find a list of centroids along WTMM lines and perform NNLS fitting to identify and quantify single or

multiple overlapped peaks. Each local maxima defines a range around its channel number and it is possible to merge close overlapping ranges to form a larger range for calculation at an average scale. The advantage is to avoid peaks whose FWHM is too small. This approach may be bad if the overlapping peaks are far away from each other and the DRF line is not flat enough.

2.4 Peak Area Uncertainty

2.4.1 Error Propagation of Non-negative Least Square Method [23–26]

Recall the fitting problem in the previous section:

$$Bk = S. \tag{2.14}$$

NNLS is based on the least square method. A sub matrix of the basis matrix is defined by setting some columns of the basis matrix to zero. The main idea of NNLS is to try different possible sub matrices until the fit vector consists of no negative components and the error is the least (see [19] for the detailed algorithm). It we denote this sub matrix as B_1, then k is determined by

$$B_1^T B_1 k = B_1^T S. \tag{2.15}$$

In MATLAB, this can be solved with $k = (B_1^T B_1) \setminus (B_1^T S)$. If the inversion of the matrix $B_1^T B_1$ exists, then

$$k = (B_1^T B_1)^{-1} B_1^T S = OS \text{ with } O = (B_1^T B_1)^{-1} B_1^T. \tag{2.16}$$

If S is uncorrelated, its covariance matrix C_S can be simply determined by

$$C_s = \sigma_S^2 I \tag{2.17}$$

$$\sigma_S = \frac{1}{m-n} S_{obj} \tag{2.18}$$

$$S_{obj} = S^T (I - H) S \tag{2.19}$$

$$H = B_1 O \tag{2.20}$$

Here $m - n$ is the freedom degree of B_1.

If not, C_S needs to be calculated from the covariance matrix of the original spectrum x. We can write the wavelet transform as a matrix form $WX = S$ and W represents the wavelet transform operation. This can be verified by the figure below, which compares the results given by the convolution and the matrix form.

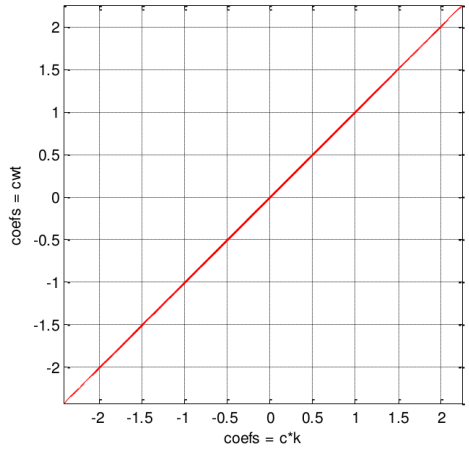


Figure 2.8: Comparison between the convolution (y-axis) and matrix form (x-axis) of wavelet transform.

If we denote the covariance matrix of the signal as $C_x = I\sigma_n^2$, then we have

$$C_s = WC_xW^T \tag{2.21}$$

$$C_k = OC_sO^T, \tag{2.22}$$

and the uncertainty of vector k is the diagonal elements of the covariance matrix C_s. Here σ_n denotes the standard deviation of noise in the spectrum. This can be estimated using the method in subsection 2.4.3.

2.4.2 Error Propagation for Singular Basis Matrix

In fact, the matrix $B_1^T B_1$ is singular for the basis matrix used in the algorithm and its inversion does not exist. The are two methods to estimate this: (1)

The Moore-Penrose pseudo inversion [27, 28]; (2) Truncated singular value decomposition (TSVD) [29, 30].

The Moore-Penrose pseudoinverse B of matrix A is defined as

$$ABA = A \tag{2.23}$$

$$BAB = B \tag{2.24}$$

$$AB \text{ is Hermitian.} \tag{2.25}$$

$$BA \text{ is Hermitian.} \tag{2.26}$$

The pesudo inversion B of matrix A can be solved based on the above constraints ($B = \text{pinv}(A)$ in MATLAB). Recall Eq. (2.16),

$$k = (B_1^T B_1)^{-1} B_1^T S = OS \text{ with } O = (B_1^T B_1)^{-1} B_1^T. \tag{2.27}$$

We can use $\text{pinv}(B_1^T B_1)$ to approximate $(B_1^T B_1)^{-1}$. The area uncertainty obtained using this method was not consistent with that given by OriginLab, indicating pseudo inversion may not work here. However, it was found in practical tests that if the result given by pseudo inversion was divided by a constant, it will be consistent with that given by OriginLab in the simulation. The reason is unknown. And this empirical rule does not work well in real spectra. For this reason, all the area uncertainties will be calculated using another method in Chapter 3 and 4: the truncated singular value decomposition (TSVD).

Singular value decomposition (SVD) is a factorization method of a matrix. It can be used to calculate pseudo inversion of singular matrix. Although the stability of SVD depends on the specific matrix to be inverted, the basis matrix in our method is constructed from Gaussian functions and DRF line, which does not depend on specific spectra. At present, this method works well for the analyzed spectra up to date. Other methods such as conjugate gradient methods and Levevberg-Marquardt methods may be worthy of testing in the future as well. For the following problem,

$$B_1 k = S \tag{2.28}$$

B_1 can be decomposed as UEV^T where U and V are two unitary matrices, V^T denotes the conjugate transpose of V and E is a diagonal matrix with non-negative real numbers on the diagonal. The diagonal entries of E are known as the singular values of B_1. The pseudo inversion of B_1 can be written as VE^+U^T where E^+ is the pseudo inversion of E which can be calculated by replacing entries on the diagonal by its reciprocal. We could set a threshold and set entries on the diagonal of E below this threshold to zero, which is called truncated singular value decomposition (TSVD). TSVD is useful for problems like the inversion of a singular matrix. To summarize, TSVD uses the following equations:

$$B_1 = UEV^T \tag{2.29}$$

$$k = VE^+U^TS \tag{2.30}$$

$$E(E < threshold) = 0 \tag{2.31}$$

2.4.3 Noise Level Estimation

Previously in Eq.(2.21), the standard deviation of the noise is used in the calculation of the area uncertainty. The method to estimate the noise level will be discussed in this subsection.

The main source of peak area uncertainty comes from the Poissonian noise in the spectrum. For white noise that does not depend on frequency, the coefficients of the wavelet transform at the centroid channel will be evenly distributed. However, there will be a maximum at the scale of the local maxima along WTMM line of the peak. Based on this, the standard deviation of noise could be estimated from the coefficients at the first few small scales. Figure. 2.9 was generated using a Gaussian function in addition with white noise and shows the .

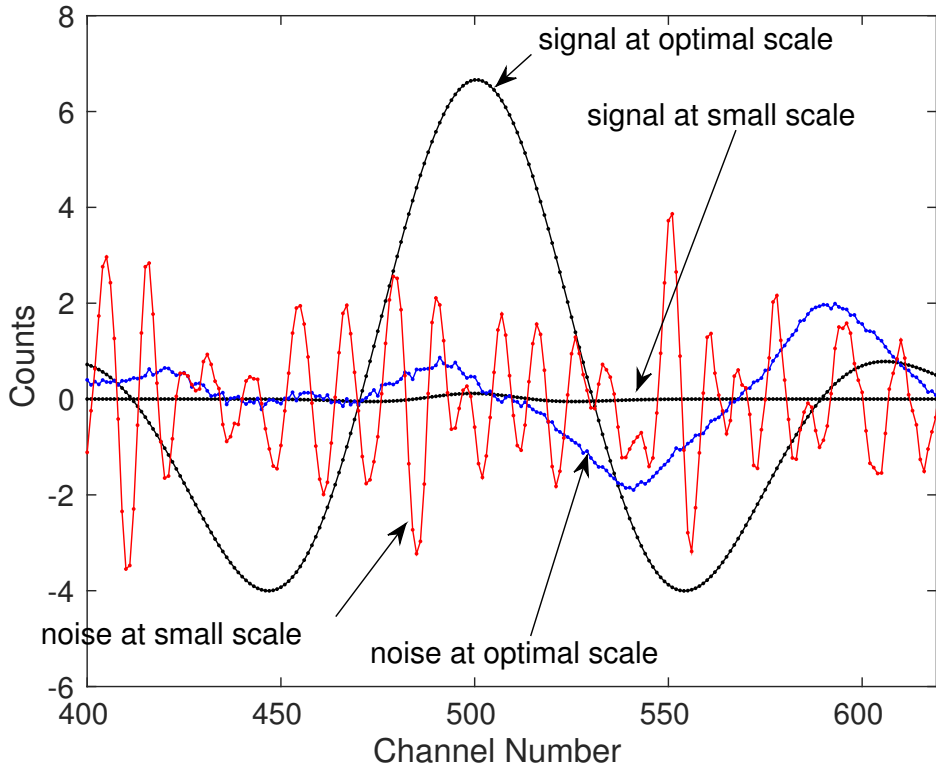


Figure 2.9: Noise level determined from coefficient at the smallest scale.

The following formula is used in the wavelet algorithm,

$$\sigma_n = 0.0501/0.0228 * std(coefs) \tag{2.32}$$

where σ_n is standard deviation of the noise, $coefs$ is coefficients at a given channel, std represents standard deviation of a vector (the same function in MATLAB) and 0.0501/0.0228 is a constant accounting for the change of σ from peak to corresponding coefficient vector. It is obtained by a large number of simulation tests. It is reasonable to assume that σ_n is constant near the peak centroid because only a limited range around the centroid is analyzed when deconvolving the true peaks in the vector k.

CHAPTER 3

SIMULATION

In this chapter, we will test the improved wavelet algorithm described previously as well as its associated problems using different simulated spectra. With proper designed simulated signals, the advantages and limits of the new algorithm can be clarified compared with different approaches. Some key parameters can be also optimized before further tests of experimental data are conducted. The wavelet code mainly consists of three parts: (1) WTMM line search in the scalogram; (2) The peak centroid and area calculation and (3) Calculation of the peak area uncertainty. Examples will be given for each part, while more focus will be placed on the boundary effect problem and peak area uncertainty analysis. For evaluation and review, a variety of problems and solutions will be discussed and compared. The choice of the suitable method may depend on the specific problem as well. So far, TSVD works well for a variety of spectra with complex Compton continua over a wide channel range. Further tests of many spectra are needed for a further verification of the best method.

3.1 Simple Test

First of all, some simple tests using Gaussian functions without continuum and noise will be given here to show the basic application of the wavelet algorithm. Because of no continuum and noise, there should be no error in the results. The fit vector k should be non-zero at peak centroids and zero elsewhere. The following tests were performed as shown in Figure 3.1:

1. Single peak without overlap;
2. Two peaks with the same area without overlap;
3. Two peaks with the same area with overlap;

4. Two peaks with different areas with overlap.

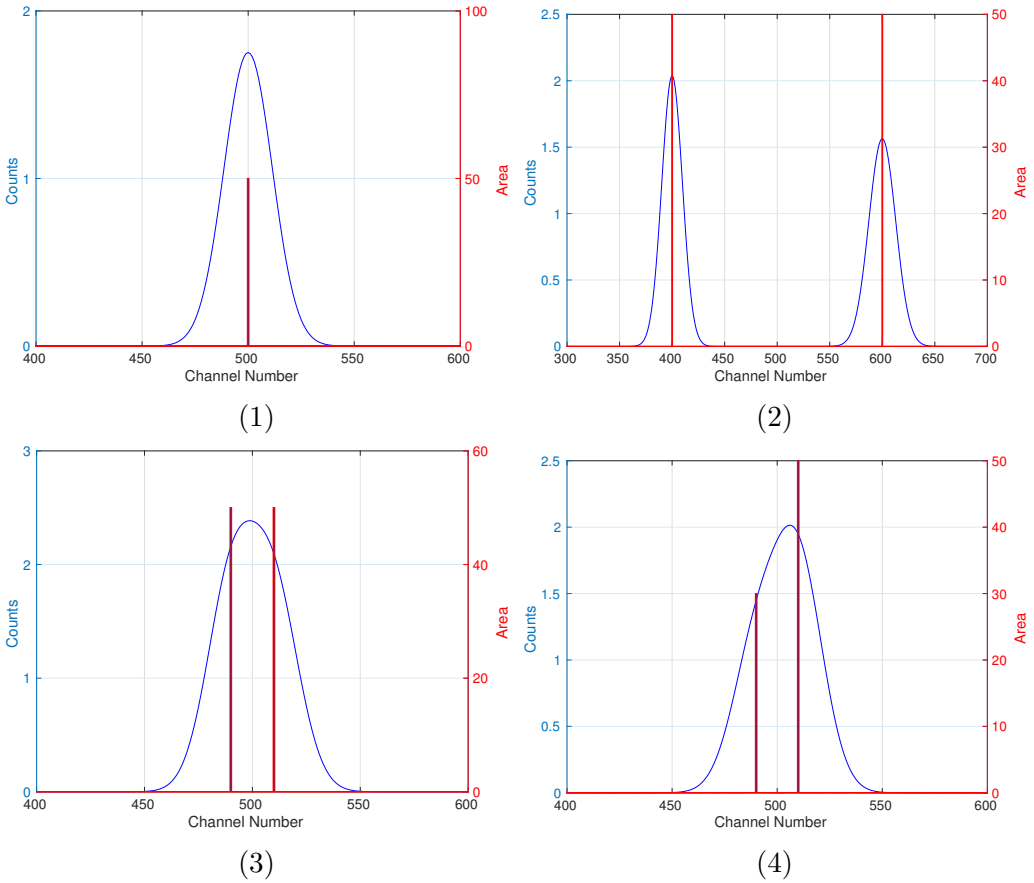


Figure 3.1: Simple simulated tests of the wavelet algorithm. The blue line represents the peak and the red bar represents the calculated peak area from the NNLS fitting.

Table 3.1: Simple simulated tests of the wavelet algorithm.

Test No.	Simulation		Calculation	
	Centroid	Area	Centroid	Area
1	500	50	500	50
2	400	50	400	50
	600	50	600	50
3	490	50	490	50
	510	50	510	50
4	490	30	490	30
	510	50	510	50

It can be seen from Table 3.1, all the peaks are found at the correct centroids and the calculated peak areas are accurate, indicating no area error. For

the last test, the two peaks are separated by 20 and there is only one local maxima in the scalogram (See Figure 3.2). As was demonstrated in Figure 2.7, there can be one or more local maxima if the distance varies. If there is one local maxima for each peak, we can perform the wavelet transform of a single peak for each local maxima. If there is only one local maxima for all the peaks, we could either perform NNLS at an average scale (e.g. the middle point of the DRF line within the region of interest. The region of interest is defined in Chapter 2.) or at the scale of the local maxima. The latter one is the scale where the effect of noise is minimized. For this simple test, both methods give the same result because there is no noise (Figure 3.4).

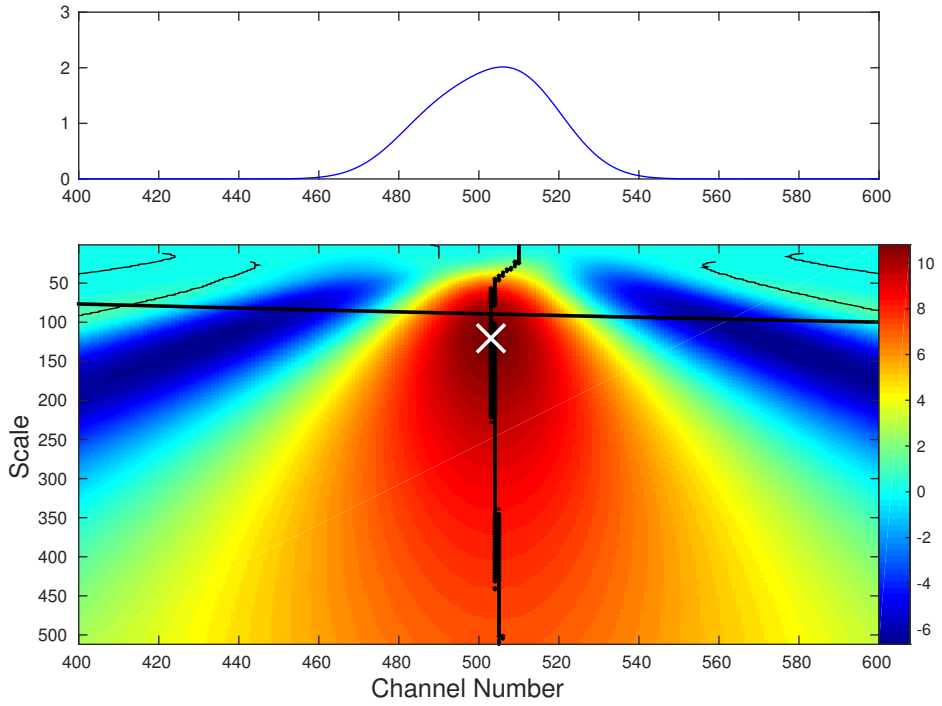


Figure 3.2: Two peaks with overlap and there is only one local maxima along the WTMM line.

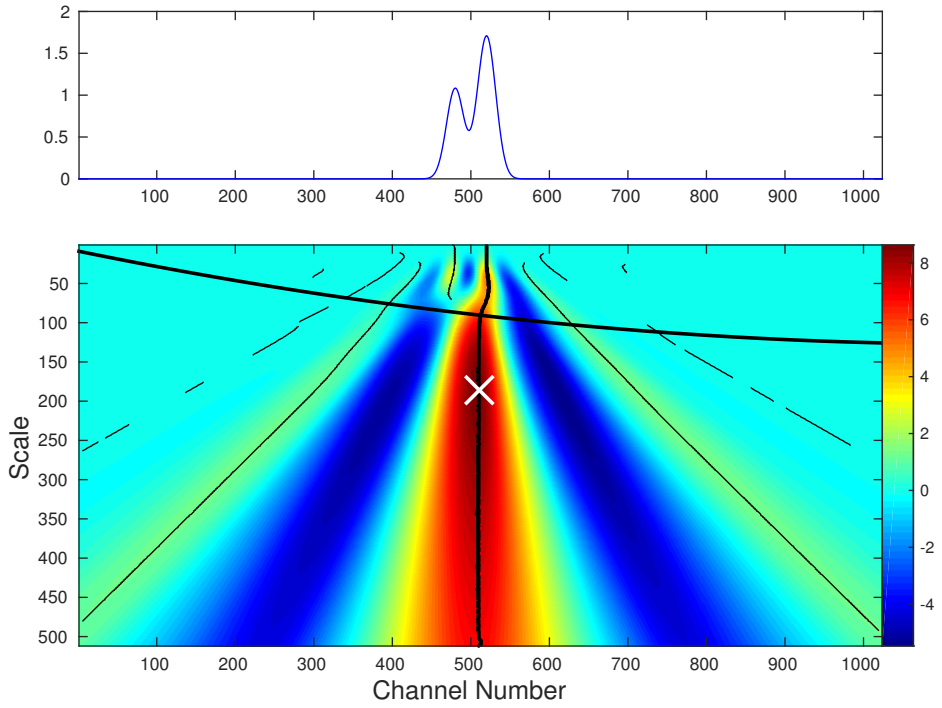
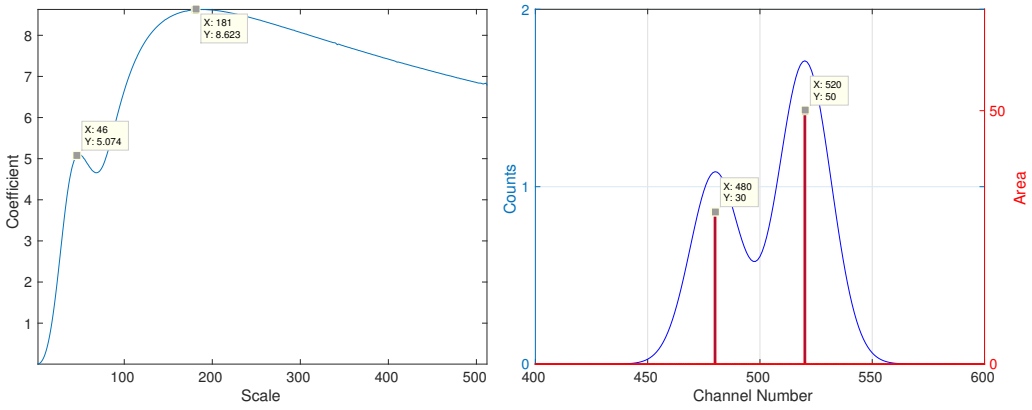


Figure 3.3: Two peaks with overlap and there are more than one local maxima along the WTMM line. Only the second local maxima below the DRF line was marked in the scalogram.

There are two local maxima along the WTMM line of the peak at channel 520 as shown in Figure 3.4a. If the local maxima is searched from the first scale ($s = 1$), the local maxima with smaller scale will be identified and this corresponds to a narrower region of interest which does not cover the other peak. As the WTMM line of the peak at channel 480 does not intersect with the DRF line, it will be ignored according to the filtering criterion. So only one peak will be identified. This indicates a scale threshold is needed to search local maxima along a WTMM line. If so, the local maxima with a larger scale will be found and both peaks will be identified because of the corresponding larger region of interest.

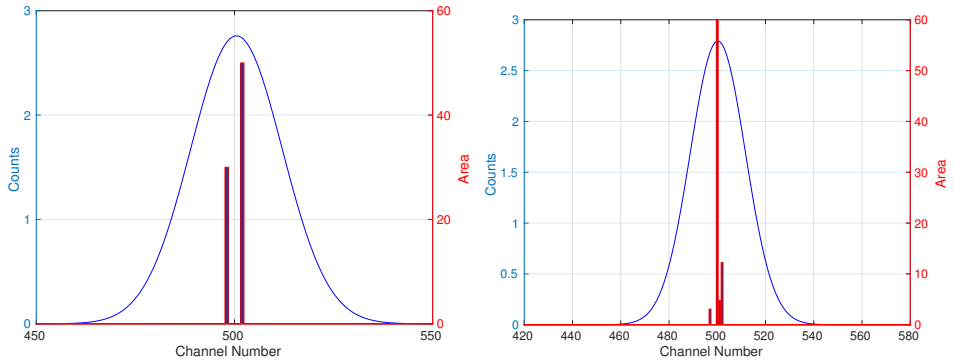


(a) The cofficient along the WTMM line.

(b) The NNLS fitting result.

Figure 3.4: Two peaks with overlap and there are more than one local maxima along a single WTMM line.

For two peaks with different areas in the simulation, it was found that the two peaks could be identified until the distance between their centroids became 2 (see Figure 3.5).

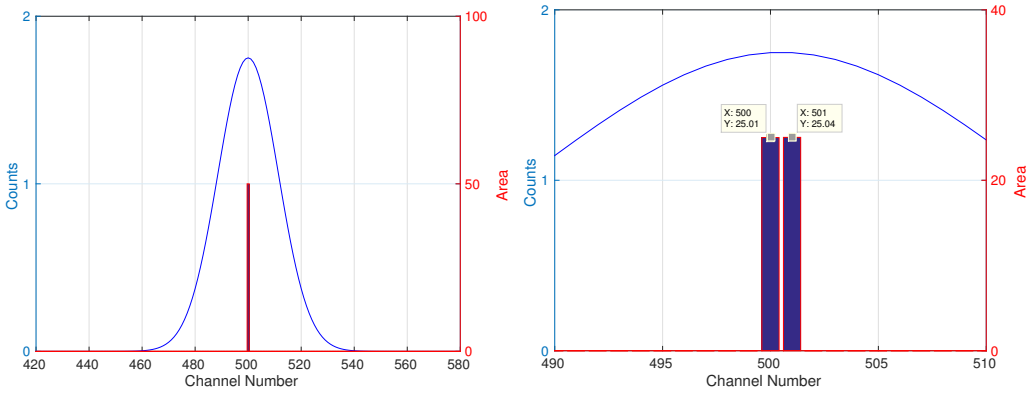


(a)

(b)

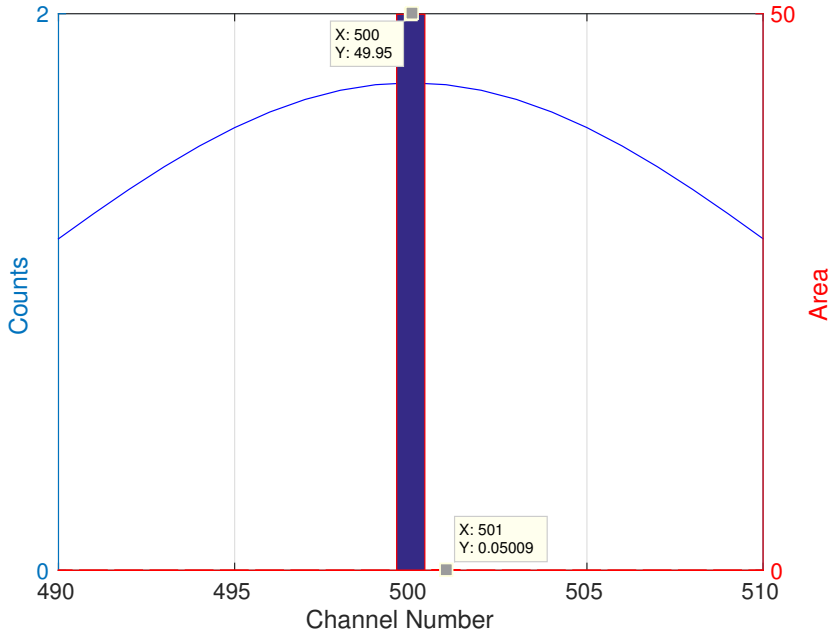
Figure 3.5: Two peaks (area 30 and 50) separated by four (left) and two (right) channels. The right one cannot be identified by the wavelet code.

The spectrum measured by the NaI detector used in this research consists of a channel range from 1 to 1024. However, peak centroids may lie between two adjacent channels. It is found in simulation that a single non-zero bin in vector k would split into multiple bins if the peak centroid channel is not a integer, as shown in Figure 3.6.



(a) $\mu = 500$

(b) $\mu = 500.5$



(c) $\mu = 500.001$

Figure 3.6: Simulation of a single peak with a non-integer centroid.

Table 3.2: Simulation of a single peak with a non-integer centroid. k_i is the i-th non-zero element in vector k, μ_i is the channel number of i-th non-zero element in vector k, μ is the calculated centroid, A is the calculated total area.

Centroid	k_1	k_2	k	μ_1	μ_2	A
500	N/A	N/A	50	N/A	N/A	500
500.5	25.0131	25.0365	50.0496	500	501	500.5002
500.001	49.9501	0.0501	50.0002	500	501	500.0010

A simple estimation would be:

$$k_1 * exp[-0.5(x-\mu_1)^2] + k_2 * exp[-0.5(x-\mu_2)^2] = A * exp[-0.5(x-\mu)^2]$$
$$k_1[1-0.5(x-\mu_1)] + k_2[1-0.5(x-\mu_2)] \approx A[1-0.5(x-\mu_0)]$$

where k_i is the i-th non-zero element in vector k, μ_i is the channel number of i-th non-zero element in vector k, μ is the calculated centroid, A is the calculated total area. So

$$A = k_1 + k_2 \tag{3.1}$$
$$\mu = (k_1\mu_1 + k_2\mu_2)/A \tag{3.2}$$

So the peak centroid is the weighted sum of the split bins' index in vector k and the peak area is the sum of the split bins' value in vector k.

3.2 Boundary Effect Problem

In this section, the method of analysis of a small region of the spectrum and its associated problems will be discussed. Note that this method is not necessary if the result given by NNLS fitting over all channels can be interpreted, which is possible with the help of the peak area uncertainty. If the whole sepctrum is used, it can be processed as below before sent to the NNLS function.

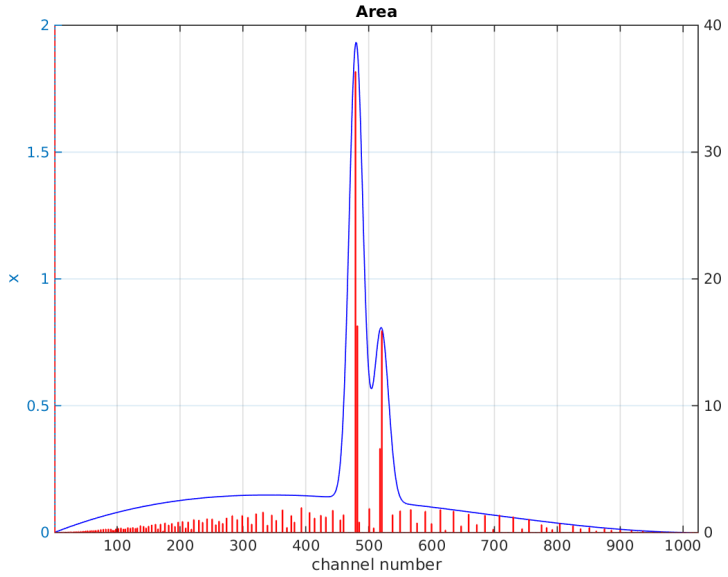
$$x = [\text{zeros}(\text{size}(x), x, \text{zeros}(\text{size}(x)]. \tag{3.3}$$

The coefficient of the wavelet transform using the "bior2.6" wavelet is zero for linear functions if the integration is taken from $-\infty$ to ∞. However, true spectra have finite length. If the whole spectrum (from channel number 1 to 1024 in our detector) is used to perform NNLS, then the continuum outside the region of interest may affect the fitting within the region of interest if the continuum varies greatly across the range of channels. A natural idea is to only perform NNLS within the region of interest and set proper values elsewhere.

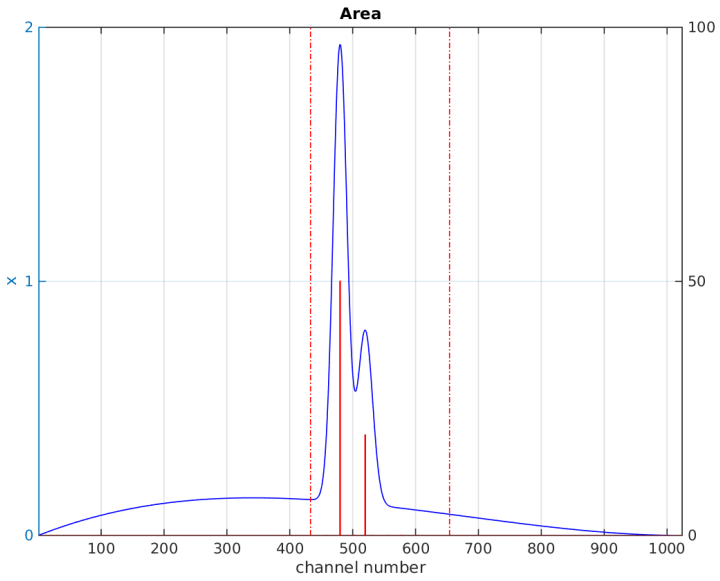
The region sent to the NNLS function can be chosen to be $3-4$ times the

peak width. If multiple peaks accounts for one local maxima, an approximation of standard deviation can be estimated from the linear relationship between the standard deviation of a peak and the corresponding scale of its local maxima based on the simulation of single peaks (see Fig. 2.5).

If only a subset of the spectrum (with zero padding outside the region of interest) is sent to the NNLS function, there will be a sharp jump on the boundary which can lead to false results in the fit vector. This effect of discontinuity on the boundary can be reduced by the continuum fitting which cannot be done when the whole spectrum is used. The continuum may vary not significantly within the small region while greatly over the whole spectrum. The difficulties are (1) The NNLS fitting result may be sensitive to the boundary used in the continuum fitting and (2) The continuum may not be linear. An improvement is observed in Figure 3.7, where a baseline was added to two peaks (area 50 and 20) with no noise. In Figure 3.7a, the peaks with baseline were sent to the NNLS function; in Figure 3.7b, the peaks with fit baseline subtracted were sent to the NNLS function.



(a) w/o subtraction



(b) w/ subtraction

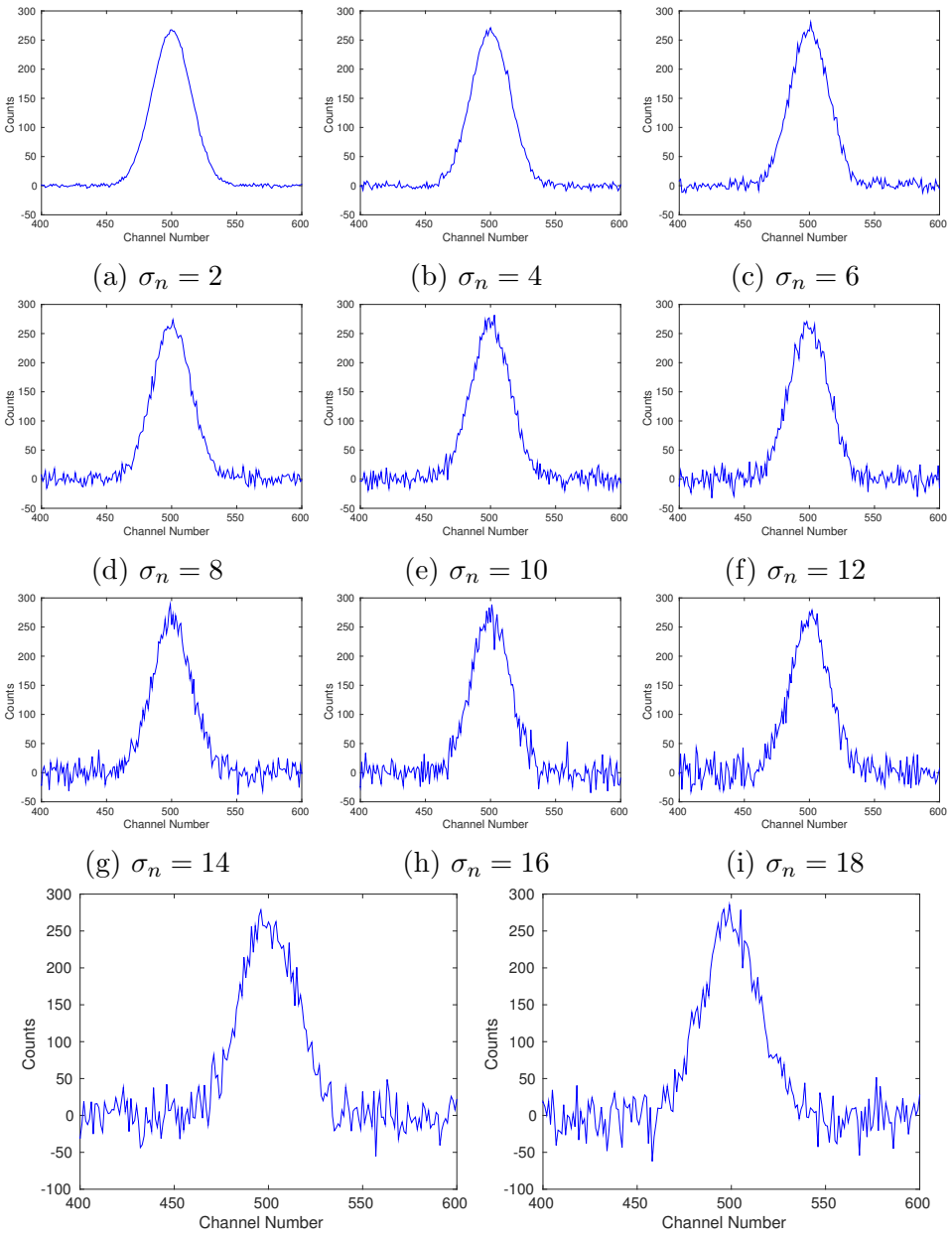
Figure 3.7: Comparisons without (left) and with (right) continuum subtraction for the simple simulation above.

3.3 Peak Area Uncertainty

In Chapter 2, two main methods are introduced to calculate the area uncertainties of photopeaks: (1) Moore-Penrose pseudo inversion and (2) truncated singular value decomposition (TSVD). In this section, these methods will be evaluated by simulation.

3.3.1 Pseudo Inversion

Pseduo inversion is a straightforward method that can be easily applied in error propagation. To prove this method actually works, a relationship between the area uncertainty and signal to noise ratio (SNR) should be shown here. Eleven Gaussian functions were tested in Figure 3.8 with peak area $A = 10000$, standard deviation of the Gaussian function $\sigma = 15$, FWHM of the Gaussian function FHWM = 35.32 and standard deviation of the noise $\sigma_n = 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22$.



(a) $\sigma_n = 2$ (b) $\sigma_n = 4$ (c) $\sigma_n = 6$

(d) $\sigma_n = 8$ (e) $\sigma_n = 10$ (f) $\sigma_n = 12$

(g) $\sigma_n = 14$ (h) $\sigma_n = 16$ (i) $\sigma_n = 18$

(j) $\sigma_n = 20$ (k) $\sigma_n = 22$

Figure 3.8: Gaussian functions with noise at different levels. The peak area is 100 and σ=15. The standard deviation of the noise is $2, 4, 6, \cdots 22$.

Even for noise with the same σ_n, the sample distribution may be different which leads to slightly different peak area uncertainty. Large tests for every σ_n were performed and an average value was used to produce Figure 3.9.

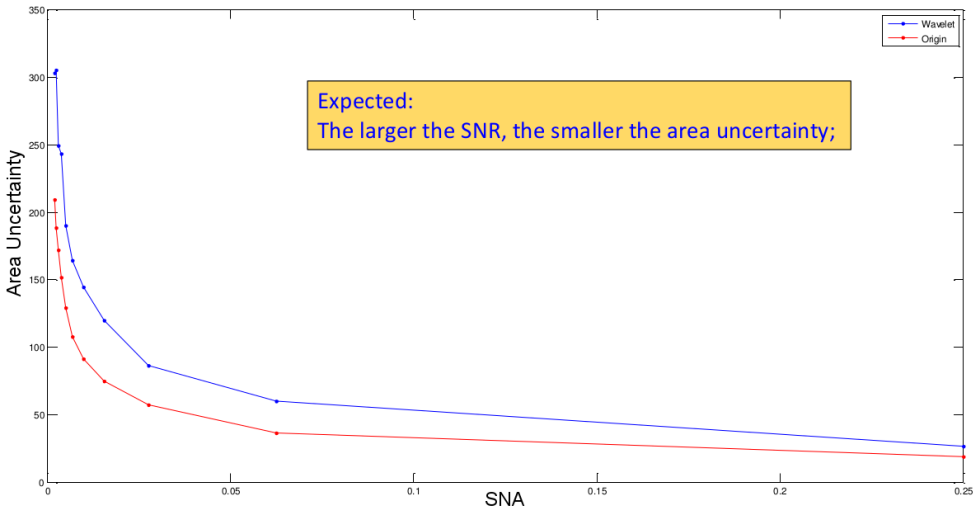


Figure 3.9: Comparison of calculated area uncertainty between NNLS (blue) and software OriginLab (red). The blue curve was calculated by the wavelet algorithm and red curve by OriginLab. Both gave similar results and it could be seen that the larger the signal to noise ratio (SNR), the smaller the area uncertainty ($\sigma_A \propto 1/\text{SNR}^2$).

3.3.2 Truncated Singular Value Decomposition

Figure 3.10 shows an example of area uncertainty calculation using TSVD for two overlapping peaks with $\mu = 480, 520$, $A = 30, 50$.

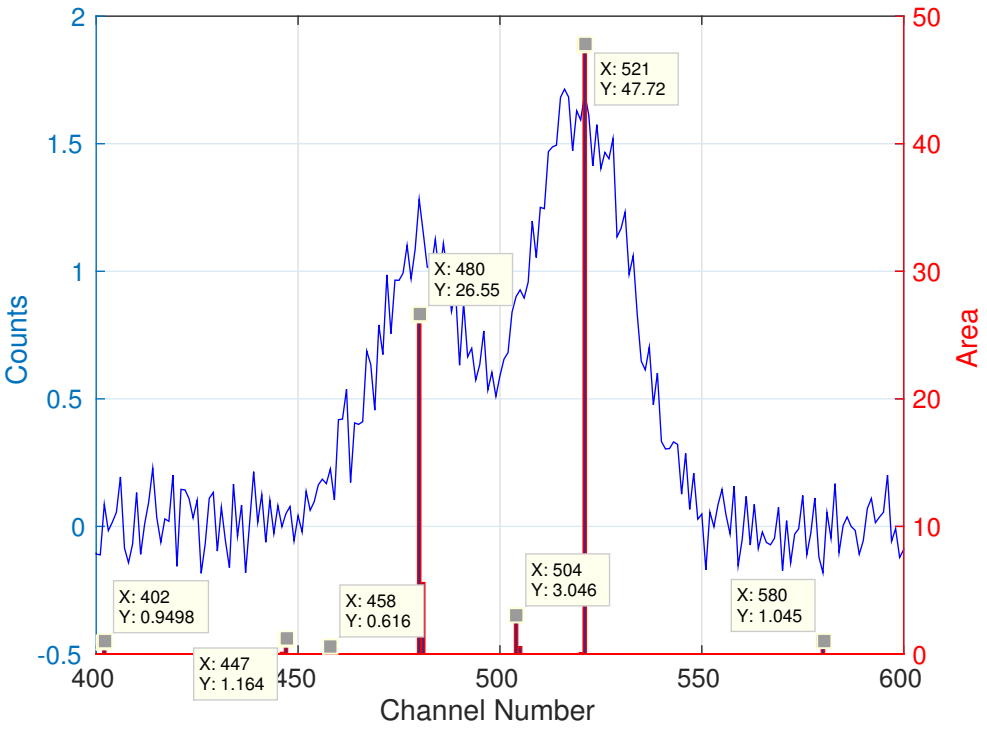


Figure 3.10: The NNLS fitting result using TSVD for two overlapping peaks with $\mu = 480, 520$, $A = 30, 50$.

Table 3.3: The NNLS fitting result using TSVD for two overlapping peaks with $\mu = 480, 520$, $A = 30, 50$.

Centroid	Area (A)	Area Uncertainty (σ_A)	σ_A/A
402.00	0.95	1.45	152.61%
446.90	1.29	1.19	92.37%
458.00	0.62	2.51	407.56%
480.17	32.15	1.39	4.33%
504.15	3.60	1.35	37.35%
521.00	47.81	1.16	2.43%
580.00	1.05	1.86	178.13%

In Figure 3.10, seven possible peaks were found. The peak at channel 504.15 which is a false peak has a substantial area (in this example) and cannot be ignored. If we assume the true peaks should satisfy $\sigma_A/A < 5\%$ (a rough estimation, 5% is chosen arbitrarily here) where σ_A is the area uncertainty and A is the peak area, only peaks at channel 480.17 and 521.00 are left. These two peaks are actually the only true peaks in the simulation. This example demonstrates that with the help of area uncertainty, the overlapping

true peaks can possibly be found.

CHAPTER 4

EXPERIMENT

Experimental spectra are more complicated than simulated Gaussians due to the presence of various background noises and the key parameters in the wavelet algorithm need to be optimized with large amounts of real spectra. Statistical results are needed to show the effectiveness of the algorithm. Significant difference exists between simulation and experiments and more focus will be put on experiments in this chapter, mainly the analysis of the peak area uncertainty.

Many possible methods are discussed in the previous chapters such as continuum fitting and zero padding. These methods may be helpful in improving the fitting quality while their fully applications need more exploration. To focus on the area uncertainty analysis, in this chapter, the whole spectrum will be sent to the NNLS function without fitting or zero padding and the area uncertainty will be calculated only using the TSVD method. Some assumptions used in the calculation: (1) Non-zero entries in the fit vector separated by certain channels are considered as one peak (~ 5); (2) There is a threshold scale to search the local maxima along a WTMM line; (4) Use a threshold to set small eigenvalues in SVD to zero (~ 0.01); (5) Peak area to area uncertainty ratio should be smaller than a certain level for true peaks (e.g. $\sigma_k/k < 5\%$).

4.1 ^{137}Cs

In Figure 4.1, four local maxima were found in the scalogram of ^{137}Cs, corresponding to the four peaks: (1) backscatter peak; (2) photopeak of ^{137}Cs at 662 keV; (3) background peak of ^{40}K at 1460 keV and (4) background

peak of ^{208}Tl at 2614 keV. The WTMM lines are not continuous and the adjacent points along the lines can be separated by several scales and several channels. To find the above four peaks, 5 scales and 5 channels are used to generate the WTMM lines.

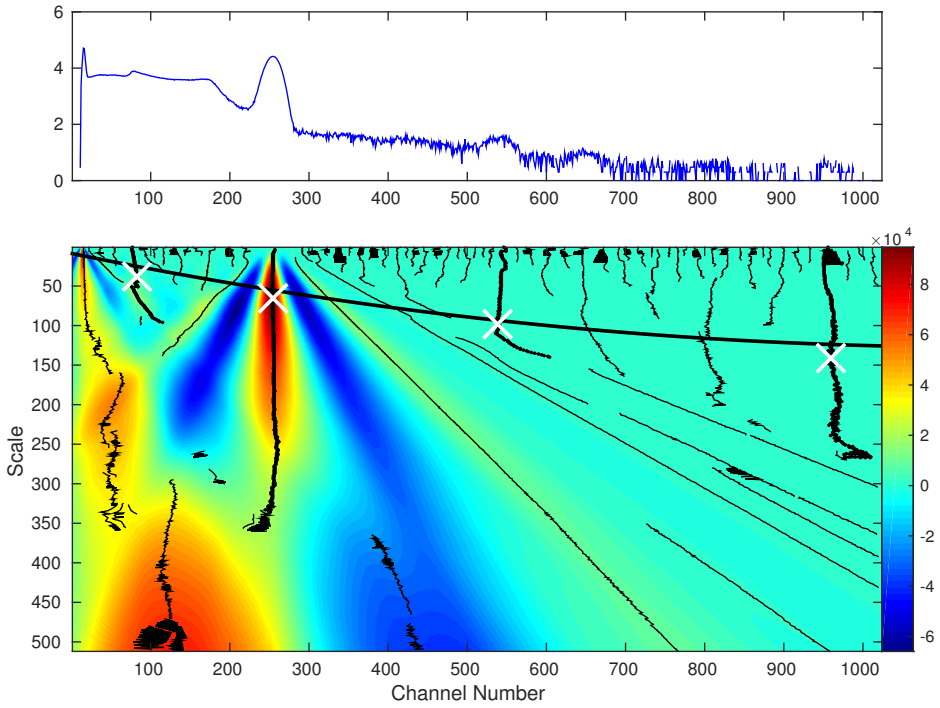


Figure 4.1: The scalogram of the wavelet analysis of ^{137}Cs.

The following tables show the NNLS fitting results of the four peaks. It should be noted that (1) The noise level used in the area uncertainty calculation is a rough estimation (2) The σ_A/A ratio is also a rough filtering criterion. However, the area uncertainty calculated using TSVD did identify the true peaks as the ones with lowest σ_A/A ratio as shown in the following tables.

The backscatter peak is identified at channel 76.04 (energy 131.84 keV) with the filtering condition $\sigma_A/A < 1\%$.

Table 4.1: The NNLS fitting result of Backscatter peak in the spectrum of ^{137}Cs.

Centroid	Energy (keV)	Area (A)	Area Uncertainty (σ_A)	σ_A/A
76.04	131.84	235681.02	636.38	0.27%
102.44	211.05	89557.99	1001.46	1.12%

The photopeak (662 keV) is identified at channel 254.55 (energy 655.02 keV) with the filtering condition $\sigma_A/A < 5\%$.

Table 4.2: The NNLS fitting result of the photopeak at 662 keV in the spectrum of ^{137}Cs.

Centroid	Energy (keV)	Area (A)	Area Uncertainty (σ_A)	σ_A/A
239.91	613.21	26897.66	1428.67	5.31%
254.55	655.02	481829.51	2081.02	0.43%
287.00	746.99	1148.15	1138.77	99.18%

The background peak of ^{40}K (1460 keV) is identified at channel 537.43 (energy 1424.34 keV) with the filtering condition $\sigma_A/A < 11\%$. The bin at channel 555.61 (1471.28 keV) seems more close to the true peak (1460 keV). But from the measured spectrum, it can be seen that the bin at channel 537.43 is more close to the actual peak (see Figure 4.2).

Table 4.3: The NNLS fitting result of the background peak of ^{40}K at 1460 keV in the spectrum of ^{137}Cs.

Centroid	Energy (keV)	Area (A)	Area Uncertainty (σ_A)	σ_A/A
490.44	1301.62	1399.13	192.10	13.73%
537.43	1424.34	1192.81	125.00	10.48%
555.61	1471.28	819.11	137.89	16.83%

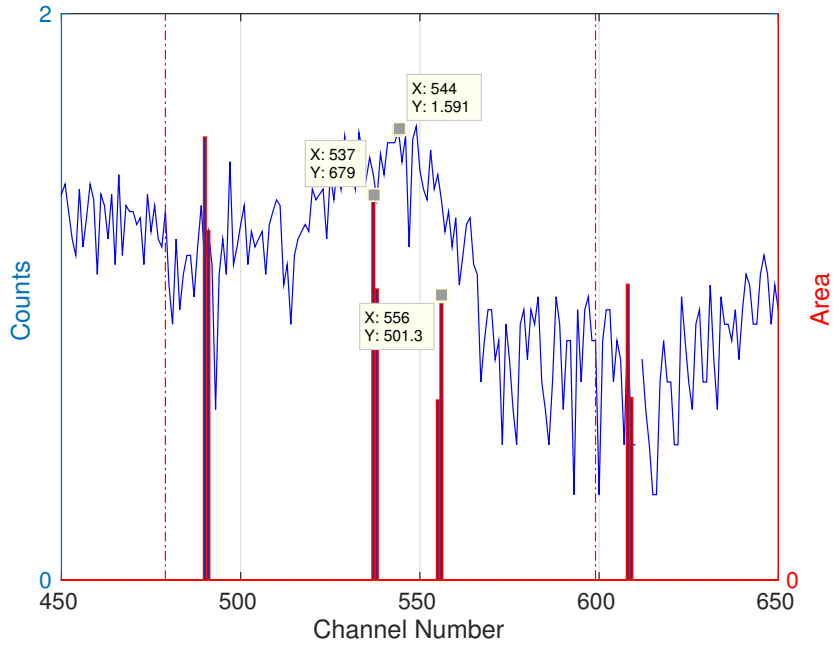


Figure 4.2: The NNLS fitting result of the background peak of ^{40}K at 1460 keV in the spectrum of ^{137}Cs.

The background peak of ^{208}Tl (2614 keV) is identified at channel 1024.00 (energy 2576.22 keV) with the filtering condition $\sigma_A/A < 5\%$.

Table 4.4: The NNLS fitting result of the background peak of ^{208}Tl at 2614 keV in the spectrum of ^{137}Cs.

Centroid	Energy (keV)	Area (A)	Area Uncertainty (σ_A)	σ_A/A
880.61	2259.29	946.00	98.50	10.41%
935.49	2382.81	35.11	25.35	72.22%
1024.00	2576.22	207.34	9.25	4.46%

For each local maxima, different σ_A/A ratios are needed to identify the true peak, whose σ_A/A ratio is the lowest among all the non-zero entries of the fit vector. The calculated area uncertainty depends on the estimated noise level, which may not be accurate enough. Besides, the σ_A/A ratio may also depends on the isotope activity. The filtering criterion based on the calculated area uncertainty needs further exploration.

4.2 ^{60}Co

There are two main photopeaks at 1173 keV and 1332 keV for ^{60}Co. Because these two peaks marginally overlap, the corresponding local maxima are slightly above DRF line. This can be seen from Figure 4.5, which is consistent with the simulation result. By setting a lower filtering limit of scale, the two peaks can be identified.

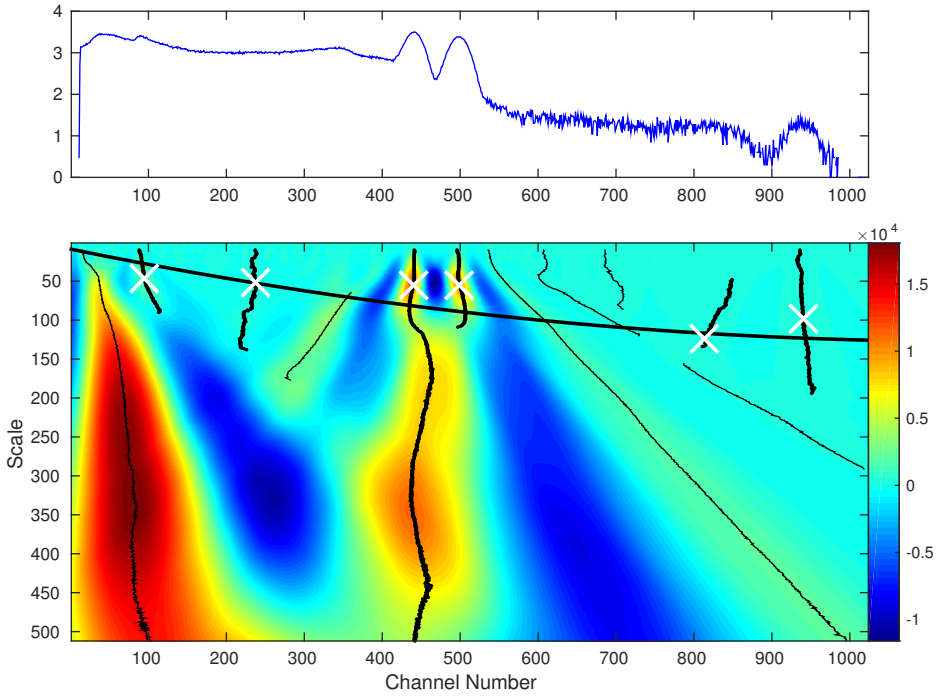


Figure 4.3: The scalogram of the wavelet analysis of ^{60}Co.

There are many other local maxima triggered in Figure 4.5. The NNLS fitting results of them are listed in Table 4.5.

Table 4.5: The NNLS fitting results of all the local maxima in the scalogram of ^{60}Co.

Centroid	Energy (keV)	Area (A)	Area Uncertainty (σ_A)	σ_A/A
95.96	191.67	126307.11	422.69	0.33%
235.40	600.29	63804.96	802.32	1.26%
418.00	1108.48	5888.24	434.81	7.26%
441.01	1170.35	84947.12	513.03	0.60%
498.10	1321.77	69485.15	3368.69	4.85%
758.00	1973.37	133.80	128.06	95.70%
782.00	2030.42	565.58	147.14	26.02%
805.00	2084.60	496.03	138.94	28.01%
830.54	2144.19	258.80	61.75	23.86%
855.48	2201.81	498.36	48.24	9.68%
890.00	2280.62	26.14	33.24	127.14%
937.08	2386.35	934.01	21.83	2.34%
981.00	2483.15	28.60	31.73	110.93%

By choosing a filtering criterion $\sigma_A/A < 5\%$, five peaks can be identified: channel 95.96, 235.40, 441.01, 498.10 and 937.08. The two photopeaks at channel 441.01 (1170.35 keV) and 498.10 (1321.77 keV) are the two main photopeaks of ^{60}Co. The photopeak at channel 937.08 (2483.15 keV) is the background peak of ^{208}Tl at 2614 keV.

4.3 ^{133}Ba

The local maxima in Figure 4.4 were searched along the WTMM lines above the DRF line. This triggered the background peak of ^{208}Tl at 2614 keV. However, the first local maxima in the scalogram is above the DRF line, which indicates a narrower region of interest compared to the local maxima below the DRF line. The problem is that some true peaks may stay outside this region of interest. The main photopeaks (276 keV, 301 keV, 356 keV and 383 keV) of ^{133}Ba overlapped in the first local maxima. Because of its importance, the first local maxima and the corresponding NNLS fitting will be analyzed separately in the following discussion. The NNLS fitting results

of the other local maxima are listed in Table 4.6. The background peak of ^{208}Tl at 2614 keV is identified at channel 777.49 (2490.83 keV).

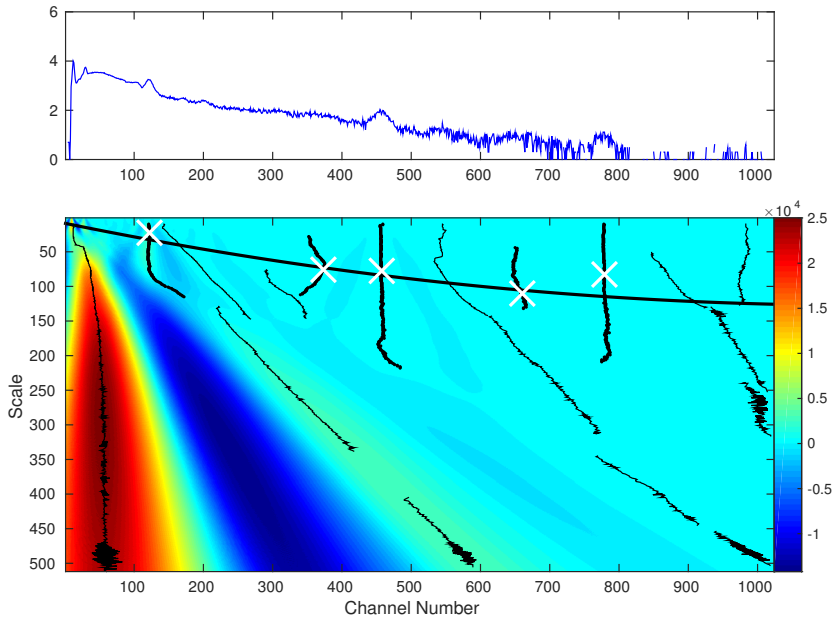


Figure 4.4: The scalogram of the wavelet analysis of ^{133}Ba. The local maxima were searched from the smallest scale.

Table 4.6: The NNLS fitting results of the wavelet analysis of ^{133}Ba except the first local maxima in Figure 4.4.

Centroid	Energy (keV)	Area (A)	Area Uncertainty (σ_A)	σ_A/A
114.87	323.48	8253.08	348.70	4.23%
119.80	339.60	15908.55	305.39	1.92%
127.11	363.51	13473.40	415.95	3.09%
134.60	388.01	5403.89	370.60	6.86%
369.50	1156.34	5482.09	123.55	2.25%
426.00	1341.15	868.66	85.29	9.82%
457.00	1442.55	2544.17	47.21	1.86%
497.00	1573.38	790.00	38.98	4.93%
610.00	1943.00	547.10	46.97	8.59%
636.00	2028.04	190.56	29.23	15.34%
665.87	2125.74	414.90	18.76	4.52%
694.95	2220.86	298.25	17.99	6.03%
749.00	2397.65	28.73	20.87	53.87%
777.49	2490.83	366.42	8.66	2.36%
809.00	2593.91	32.80	21.23	64.72%

Figure 4.5 shows the searched local maxima by setting a threshold scale along the WTMM lines. The background peak of ^{208}Tl was not triggered and the first local maxima was below the DRF line. This indicates a wider region of interest. The NNLS fitting results of the first local maxima above and below the DRF line are compared in Figure 4.6.

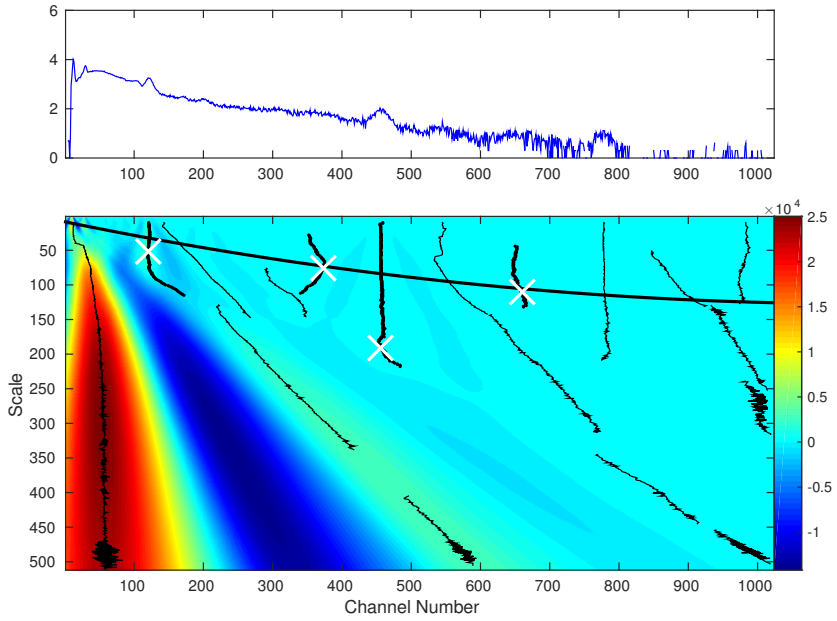
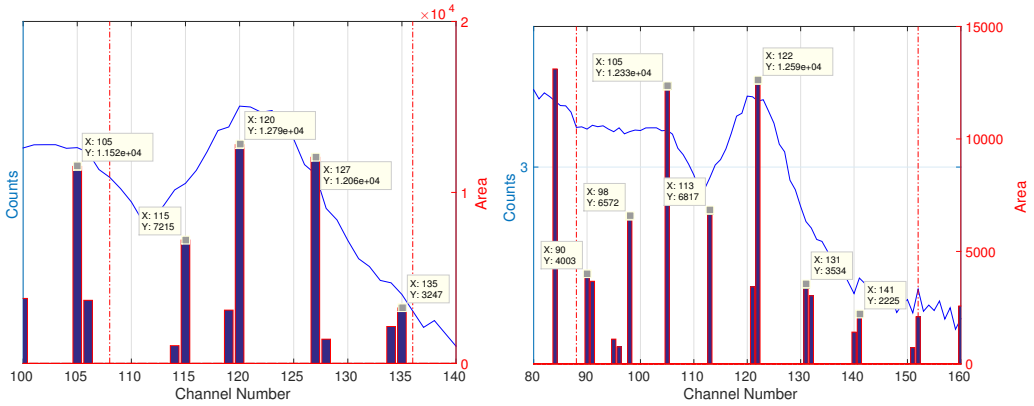


Figure 4.5: The scalogram of the wavelet analysis of ^{133}Ba. The local maxima were searched from the threshold scale.



(a) The local maxima were searched from the smallest scale.

(b) The local maxima were searched from the threshold scale.

Figure 4.6: The NNLS fitting result of the first local maxima in the scalogram of ^{133}Ba.

The linear energy calibration for this measurement is

$$E = 3.2709ch - 52.253, \tag{4.1}$$

where ch is the channel number and E is the energy. Using Eq. (4.1), channel 105 corresponds to energy 291.19 keV, which is inside the region of interest

of Figure 4.6b while outside of that of Figure 4.6a. Recall that the four main peaks of ^{133}Ba are 276 keV, 301 keV, 356 keV and 383 keV, 291.19 keV should be considered as a possible true peak. So the threshold scale is necessary to get the correct local maxima. Table 4.7 shows the NNLS fitting result of Figure 4.6b.

Table 4.7: The NNLS fitting results of the first local maxima in Figure 4.5.

Centroid	Energy (keV)	Area (A)	Area Uncertainty (σ_A)	σ_A/A
97.43	266.43	8434.88	404.35	4.79%
105.00	291.19	12329.54	966.98	7.84%
113.00	317.36	6817.36	276.53	4.06%
121.79	346.11	16014.42	183.53	1.15%
131.46	377.74	6563.32	589.91	8.99%
140.61	407.67	3639.57	272.48	7.49%
151.74	444.07	2817.74	265.95	9.44%

By choosing a filtering criterion of $\sigma_A/A < 5\%$, three photopeaks can be identified from Table 4.7: channel 97.43 (266.43 keV), channel 113.00 (317.36 keV) and channel 121.79 (346.11 keV). The photopeak at channel 131.46 (377.74 keV) is very close to the true peak 383 keV, but it cannot be identified here because of its relatively large σ_A/A ratio. There are two possible reasons for this: (1) The DRF line has a relatively large slope within this region and an average scale used in the NNLS fitting may be not good enough; (2) This region of interest is wide and the assumption of a uniform noise level may be not good enough.

In summary, the wavelet algorithm using TSVD can be a powerful tool for peak centroid and area calculation from complicated spectra. Some assumptions or estimations are used only for a quick evaluation. These are:

1. The noise level is assumed constant within the region of interest. This may not be true if the region of interest is too wide. Besides, the simple estimation using the coefficient at small scales may need better alternative improvement;

2. An average scale is used in the NNLS fitting for a single local maxima corresponding to multiple overlapping peaks. The problem is that the

effect of noise may not be minimized for all the peaks at the same time;

3. Peaks separated by small number of channels are considered as one peak, as the wavelet code cannot resolve peaks that are too close;

4. The threshold of eigenvalues in SVD needs more test;

5. If no threshold is used to search the local maxima along WTMM lines and all local maxima are required to stay below the DRF line, some true peaks may be missed because the first found local maxima may stay above the DRF line in the scalogram (e.g. the two main peaks of ^{60}Co. The chosen threshold needs more tests as bad choice may trigger unnecessary peaks;

6. A simple limit on area uncertainty to area ratio is used to choose the true peaks from the fit vector. A uniform limit may only work within a narrow region. It may depend on the counting time and source activity as well.

CHAPTER 5

CONCLUSIONS AND FUTURE WORK

5.1 Conclusions

An improved method based on the continuous wavelet transform and non-negative least squares fitting and the peak area uncertainty problem have been discussed in this thesis. Possible options including boundary effect solutions, continuum subtraction and area uncertainty application have been compared and discussed. For all the simulation and spectra analyzed, the wavelet algorithm using TSVD worked well, which can be used to analyze spectra with complicated continuum and overlapping peaks. Different from traditional Gaussian fitting, the wavelet algorithm analyzes coefficients at the optimal scale and avoids the high frequency noise. So far, rough assumptions are made for certain steps (e.g. estimation of noise level) while the effectiveness in automated isotope identification of the wavelet algorithm is promising. There are four parameters that can be adjusted in the code: (1) Lower scale limit allowed for local maximal above DRF line; (2) Minimum distance between two peaks (currently set at 5); (3) Area uncertainty / Area ratio for true peaks (currently set at 5%); (4) Threshold defined in TSVD (currently set at 0.01). The future work will be focused on more tests of experimental spectra for these parameter optimizations and reliability and stability of the code for different complicated situations.

5.2 Future Work

Here are some possible suggestions that can be tried in the future to improve the code.

1. Use different scales instead of DRF to construct the basis matrix, such

as the use of a constant scale or a constant scale within the region of interest;

2. Optimize the scale threshold used for WTMM line filtering: (1) Change the scale threshold and test with different experimental spectra; (2) Test if this threshold value is independent for a single peak and (3) Test if this threshold value depends on the distance between overlapping peaks;

3. Optimize the region of interest used to find true photopeaks. The region of interest is $3-4$ times the peak length. The region should not be too wide as it may result in including additional peaks in complicated spectra. Use different lengths and compare the results;

4. Test if the area uncertainty calculated from the error propagation in NNLS fitting is well enough or if other uncertainty sources are needed to be considered;

5. Find better ways to estimate to the noise level;

6. Optimize the partial spectrum sent to the NNLS function especially when two peaks marginally overlap when a larger range may be better;

7. Optimize the minimum distance d that separates two peaks;

8. Optimize the σ_A/σ ratio;

9. Optimize the threshold in TSVD;

10. Combination of continuum fitting and TSVD;

11. Alternative ways to calculate the peak area uncertainty. The main difficulty comes from a reliable estimation of the inversion of the basis matrix. Find other possible ways to estimate the covariant matrix of fit vector k;

12. Check to what is the wavelet algorithm sensitive such as DRF, continuum and boundary location;

13. More tests of experimental spectra.

The above work may lead to a better combination of the current methods discussed in the previous chapters. Further tests can lead to optimized parameters used in the wavelet code.

APPENDIX A

WAVELET CODE STRUCTURE

Here is the structure of the MATLAB code of the wavelet algorithm.

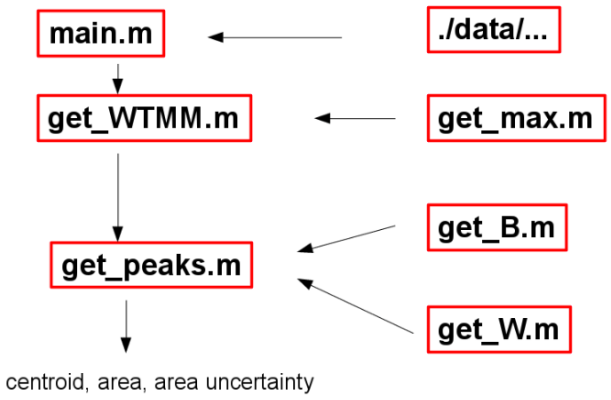


Figure A.1: Code structure of the wavelet algorithm.

Here are explanations of roles of the files:

1. main.m: The main entry of the code for initialization and configuration. Detector parameters, sepctral data and calculation options can be set in this file;

2. data: The directory to store spectra data and other resources (e.g. basis matrix);

3. get_WTMM.m: Returns a list of local maxima in the scalogram. Wavelet Transform Modulus Maxima (WTMM) line filtering is done in this function;

4. get_max.m: Returns a list of local maxima as a vector;

5. get_peaks.m: Returns peak centroids, areas and area uncertainties. NNLS is performed in this step;

6. get_B.m: Constructs the basis matrix;

7. get_W.m: Constructs a matrix that represents the wavelet transform operations.

Every time the code is run, it will store a newly-constructed basis matrix in the data directory and this will improve speed if the same base matrix is needed the next time. The MATLAB code of the most important functions have been included in Appendix B.

APPENDIX B

WAVELET ALGORITHM IN MATLAB

main.m

```
% ------------------------------------------------------------------------
% Main Entry
% Created: 05/12/14
% Revised: 10/09/14,11/13/14,11/16/14,04/02/15,04/14/15
% ------------------------------------------------------------------------
clear;clc;close('all');    % Clear all variables, command window and close
                           % all plots

% Init ------------------------------------------------------------------
% Create directories if not exist
dirs = {'data','data/base_matrix'};
for i = 1:length(dirs)
  if ~exist(dirs{i},'dir')
    mkdir(dirs{i});
  end
end

para.fDRF_s    = @(ch)-8.85460000000007e-05*ch.^2+0.20488*ch...
                 +8.80490000000001;    % scale of DRF
para.fDRF_w    = @(ch)-1.15893617021277e-05*ch.^2+0.0264238297872341*ch...
                 +1.07259574468085;    % sigma of DRF
% channel to energy calibration
% para.fE       = @(ch)3.2709*ch-52.253;    % linear @test001
% para.fE       = @(ch)2.741185488*ch-41.7212570261;    % linear @test002
para.fE        = @(ch)-0.0004578047*ch.^2+3.0821676779*ch...
                 -99.879547711;    % quadratic @test002

para.ch        = [1,1024];    % channel range
para.scale     = [1,512];     % scale range
```

```
para.wname     = 'bior2.6';   % wavelet name
% scale to sigma conversion
para.s2s.fun   = @(x)0.127866900711662*x-0.0369269493596491;
% scale to sigma conversion: sigma = c0+c1*scale
para.s2s.c0    = -0.0369269493596491;
% scale to sigma conversion: sigma = c0+c1*scale
para.s2s.c1    = 0.127866900711662;
para.s2w       = 2*sqrt(-2*log(0.5));    % 2.3548
para.INF       = 1e7;
% peaks separated by these channels are considered as one peak
para.unc_d     = 15;
para.unc_ratio = 1;
para.centroid  = [];   % peak centroid
para.area      = [];   % peak area
para.area_unc  = [];   % peak area uncertainty
ch = reshape(para.ch(1):para.ch(2),[],1);   % channel number

tmp = load('data/test001.mat');
para.x = tmp.x;
% if isfield(tmp,'x_bg')
%   para.x = para.x-tmp.x_bg;
% end

% Simulation ------------------------------------------------------------
if ~isfield(para,'x')
  len_ch = length(ch);
%   mu     = [200,460,490,510,800];
  mu = [480,520];
  sigma  = para.fDRF_w(mu);
  area   = [30,50];
  x      = 0;
  for i = 1:length(mu)
    x = x+area(i)*normpdf(ch,mu(i),sigma(i));
  end
% Add noise to simulation -----------------------------------------------
%   sigma_n = 0.1;
%   para.sigma_n = sigma_n;
%   noise = sigma_n*randn(len_ch,1);
%   save('data/noise.mat','noise');
%   load('data/noise.mat');
%   para.noise = noise;
%   x = x+noise;
  x = x+1/1024^3*(ch-ch(1)).*(ch-ch(end)).^2;
```

```
%   para.sigma = sigma;
  para.x = x;
end

% Calculate ------------------------------------------------------------------
opt.bPlot    = 1;    % Plot spectrum and scalogram
opt.bPlotLog = 1;    % Plot y-axis as log scale
opt.bPlotAll = 1;    % Plot all WTMM lines
opt.bPlotE   = 0;    % Plot x-axis as eneryg
opt.bDRF     = 1;    % Use DRF to filter
opt.err_DRF  = para.INF;   % DRF error
para.opt     = opt;  % Pass options to para struct for get_WTMM control

para = get_WTMM(para);        % Find WTMM lines in scalogram
% para.peaks = para.peaks(1,:); % Choose WTMM lines for calculation

opt = [];
opt.bPlot    = 1;    % Plot
opt.bPlotLog = 1;    % Plot in log scale
opt.bPlotE   = 0;    % Plot x-axis as eneryg
opt.bCut     = 0;    % Use signal in [mu-width*sigma, mu+width*sigma]
opt.bBase    = 0;    % Substract calculated baselin
para.opt     = opt;

% Find peaks with centroid, area and area uncertainty
para = get_peaks(para);

fprintf('Peaks found:\n');
fprintf('\tCentroid     Area            Uncertainty         Ratio\n');
centroid = para.centroid;
area = para.area;
area_unc = para.area_unc;
for i = 1:length(centroid)
  fprintf('\t%-20.2f%-20.2f%-20.2f%-20.4f\n',centroid(i),area(i),...
          area_unc(i),area_unc(i)/area(i));
end
```

get_WTMM.m

```
function para = get_WTMM(para)
% ------------------------------------------------------------------------
```

```
% Get peaks
% 05/12/14
% --------------------------------------------------------------------

fprintf('Get WTMM lines ...\n');

ch       = reshape(para.ch(1):para.ch(2),[],1);
x        = para.x;
fDRF     = para.fDRF_s;
scales   = para.scale(1):para.scale(2);
wname    = para.wname;
bPlot    = para.opt.bPlot;
bPlotLog = para.opt.bPlotLog;
bPlotAll = para.opt.bPlotAll;
bPlotE   = para.opt.bPlotE;
bDRF     = para.opt.bDRF;
err_DRF  = para.opt.err_DRF;

coefs = cwt(x,scales,wname);

if bPlot
  figure('Name','get_peaks');clf;
  scrsz = get(0,'ScreenSize');
  w = 800;h = 600;
  set(gcf,'Position',[(scrsz(3)-w)/2,(scrsz(4)-h)/2,w,h]);
  x_axis = ch;
  tmp_x  = x;
  if bPlotLog;tmp_x = log10(abs(x));end
  if bPlotE;x_axis = para.fE(ch);end
  para.h1 = subplot(3,1,1);hold on;box on;
  plot(x_axis,tmp_x,'b');
  xlim([x_axis(1) x_axis(end)]);
  para.h2 = subplot(3,1,2:3);hold on;box on;
  imagesc(x_axis,scales,coefs);
  axis([x_axis(1),x_axis(end),scales(1),scales(end)]);
  set(gca,'YDir','reverse');
  set(gca, 'Layer', 'top');
  p = get(gca,'Position');

  colormap jet;
  colorbar('Position',[p(1)+p(3),p(2),0.03,p(4)]);

  if bDRF
```

```
46      plot(x_axis,fDRF(ch),'k-','LineWidth',2);
47    end
48  end
49
50  [ns,nt] = size(coefs);
51  w = 5;h = 5;
52  flag_coefs = zeros(size(coefs));
53  len_WTMM = 0;
54  max_WTMM = 10000;
55  WTMM = cell(1,max_WTMM);
56  peaks = zeros(max_WTMM,3);
57  for i = 10:ns
58    [~,loc] = get_max(coefs(i,:));
59    for j = 1:length(loc)
60      for k = 1:h
61        tmp = flag_coefs(max(i-k,1):min(i,ns),max(loc(j)-w,1):min(loc(j)+w,nt));
62        n = sum(sum(tmp));
63        if n
64          n = tmp(tmp~=0);
65          n = n(1);
66          break;
67        end
68      end
69      if n == 0
70        len_WTMM = len_WTMM+1;
71        n = len_WTMM;
72      end
73      flag_coefs(i,loc(j)) = n;
74      WTMM{n} = [WTMM{n};loc(j),i,coefs(i,loc(j))];
75    end
76  end
77
78  buf = cell(1,len_WTMM);
79  len_buf = 0;
80  ss = {'r.-','b.-'};
81
82  % Filtering ----------------------------------------------------------------
83  for i = 1:len_WTMM
84    p = WTMM{i};
85    if p(1,1) > 371 && p(1,1) < 373
86  %     fprintf('DEBUG ...')
87    end
88    % Length check
```

```
89    if length(p) <= 70
90  %     continue;
91    end
92
93    if bPlotAll
94      plot(x_axis(p(:,1)),p(:,2),'k');%ss{2-mod(i,2)});
95    end
96
97    % Cross check
98  %   k = max(5,fix(0.1*size(p,1))); % If cross too close at the edge
99    k = 0;
100   a = p(1+k,:);b = p(end-k,:);
101   if (fDRF(a(1))-a(2))*(fDRF(b(1))-b(2)) > 0
102     continue;
103   end
104   [~,loc_cross] = min(abs(p(:,2)-fDRF(p(:,1))));
105
106   % Slope check
107   w = max(p(:,1))-min(p(:,1));
108   h = max(p(:,2))-min(p(:,2));
109   if w>0.5*h
110     continue;
111   end
112
113 %   Maxima check
114   [~,loc] = min(abs(p(:,2)-fDRF(p(1,1))));
115   loc_s = max(1,p(loc,2)-err_DRF);
116   [~,loc] = min(abs(p(:,2)-loc_s));
117 %   loc = 1;
118   if isempty(get_max(p(loc:end,3)))
119     continue;
120   end
121   [~,loc1] = get_max(p(loc:end,3));
122   k1 = loc1(1)-1+loc;k = k1;
123   smax = p(k,2);
124   if bDRF && smax < fDRF(p(k,1))-err_DRF
125 %       || (smax > fDRF(p(k,1)) && abs(p(loc_cross,1)-p(k,1)) > 10)
126     continue;
127   else
128     if bPlot
129       plot(x_axis(p(:,1)),p(:,2),'k.');
130       plot(x_axis(p(k,1)),p(k,2),'wx','MarkerSize',20,'LineWidth',2);
131     end
```

```
132     len_buf = len_buf+1;
133     buf(len_buf) = WTMM(i);
134     peaks(len_buf,:) = p(k,:);
135   end
136 end
137
138 if bPlot
139   if bPlotE
140     xlabel('Energy ( keV )');
141   else
142     xlabel('Channel Number');
143   end
144   ylabel('Scale');
145 end
146
147 WTMM = buf;
148 len_WTMM = len_buf;
149 WTMM(len_WTMM+1:end) = [];
150 peaks(len_WTMM+1:end,:) = [];
151 [~,tmp] = sort(peaks(:,1));
152 peaks = peaks(tmp,:);
153 WTMM  = WTMM(tmp);
154
155 para.WTMM  = WTMM;
156 para.peaks = peaks;
157 para.coefs = coefs;
```

get_peaks.m

```
1  function para = get_peaks(para)
2  % ------------------------------------------------------------------------
3  % Get peaks with centroid, area and area uncertainty
4  % 05/12/14
5  % 10/09/14,11/13/14
6  % ------------------------------------------------------------------------
7
8  fprintf('Get peaks with centroid and area ...\n');
9
10 ch       = reshape(para.ch(1):para.ch(2),[],1);
11 x        = para.x;
12 peaks    = para.peaks;
```

```
13  s2s_fun  = para.s2s.fun;
14  wname    = para.wname;
15  fDRF_s   = para.fDRF_s;
16  fDRF_w   = para.fDRF_w;
17  len_ch   = length(ch);
18
19  bPlot    = para.opt.bPlot;
20  bPlotLog = para.opt.bPlotLog;
21  bCut     = para.opt.bCut;
22  bBase    = para.opt.bBase;
23
24  err = 1e-5;
25  width = sqrt(-2*log(err));
26  fprintf('\twidth = %.4f\n',width);
27  width = 4;
28  % p = check_peaks(peaks,s2s_fun,width,len_ch);
29  np = size(peaks,1);
30  for i = 1:np
31  %   ind1 = p(i,1);
32  %   ind2 = p(i,2);
33  %   mu = (ind1+ind2)/2;
34  %   sigma = fDRF_w(mu);
35    mu = peaks(i,1);
36    smax = peaks(i,2);
37    if isfield(para, 'sigma')
38      sigma = para.sigma(i);
39    else
40      sigma = s2s_fun(smax);
41    end
42    ind1 = max(floor(mu-width*sigma),1);
43    ind2 = min(ceil(mu+width*sigma),len_ch);
44  %   sopt = fix(fDRF_s(mu));
45    sopt = fix(smax);
46    x1 = get_x1(x,ind1,ind2,len_ch,bCut,bBase);
47    if bPlot
48      figure('Name','debug');
49      plot(ch,x1(len_ch+1:2*len_ch),'b.-',ch(ind1:ind2),x(ind1:ind2),'ro');
50      legend('x1','x');axis('tight');
51      xlabel('ch');ylabel('x');title('x1 vs. x');
52    end
53    % ------------------------------------------------------------------------
54    fprintf('\tmu = %.4f\tsigma = %.4f\tsopt = %.4f\tind1 = %d\tind2 = %d\n',...
55      mu,sigma,sopt,ind1,ind2);
```

```
%   if sigma <= 3B1'*B1
%     tmp = fix(10/sigma*smax);
%   else
%     tmp = sopt;
%   end
  tmp = cwt(x1,sopt,wname)';
  S  = tmp(len_ch+1:2*len_ch);
  B = get_B(len_ch,fDRF_w(1:len_ch),sopt,wname);
  fprintf('Performing NNLS ...\n');
  [k,~] = lsqnonneg(B,S);%,optimset('TolX',0.1));
  % ------------------------------------------------------------------------
  if bPlot
    figure('Name','debug');hold on;box on;
    if bPlotLog
      [ax,h1,h2] = plotyy(ch,log10(x1(len_ch+1:2*len_ch)),ch,k,'plot','bar');
    else
      [ax,h1,h2] = plotyy(ch,x1(len_ch+1:2*len_ch),ch,k,'plot','bar');
    end
    box on;grid on;
    set(h1,'Color','blue');
    set(h2,'edgecolor','red');
    xlabel('Channel Number');
    ylabel('Counts');
    set(ax(2),'ycolor','r');
%     for i1 = 1:length(ax)
%       set(ax(i1),'xlim',[ch(1) ch(end)]);
%     end
  set(ax,'xlim',[1 1024])
    plot([ind1 ind1],ylim,'r-.');
    plot([ind2 ind2],ylim,'r-.');
%     axis('tight');
%     title('Area');
    set(get(ax(2),'ylabel'),'String','Area');
    set(ax,'FontSize',14);
  end

  % Calculate area uncertainty using TSVD, truncated < 0.01
  if isfield(para,'sigma_n')
    sigma_n = para.sigma_n;
  else
    sigma_n = 0.0501/0.0228*std(para.coefs(1,ind1:ind2));
  end
%   sigma_n = sigma_n+sigma_b;
```

```
99    I = k~=0;
100   B1 = zeros(size(B));
101   B1(:,I) = B(:,I);
102   [U,E,V] = svd(B1);
103   E(E<0.01) = 0;
104   O = V*pinv(E)*U';
105   %O = pinv(B1'*B1)*B1';
106   W = get_W(x,sopt,wname);
107
108   Cx = eye(len_ch)*sigma_n^2;
109   Cs = W*Cx*W';
110
111 %   H = B1*O;
112 %   Sobj = S'*(eye(len_ch)-H)*S;
113 %   sigma_s = Sobj/(length(S)-rank(B1));
114 %   Cs = eye(len_ch)*sigma_s^2;
115
116   Ck = O*Cs*O';
117   unc = sqrt(diag(Ck));%/sopt;
118
119   figure;bar(unc);
120
121   [centroid,area,area_unc,err] = ...
122     weight(x,k,ind1,ind2,para.unc_d,unc,para.unc_ratio);
123   if(sum(x(ind1:ind2)) < 0.5*area)
124     warning('x << area detected at channel %.2f\n', centroid);
125     continue;
126   end
127   if ~err
128     para.centroid = [para.centroid;centroid];
129     para.area = [para.area;area];
130     para.area_unc = [para.area_unc;area_unc];
131   end
132 end
133
134 end % End get_peaks
135
136 function x1 = get_x1(x,ind1,ind2,len_ch,bCut,bBase)
137 % Pre-process spectrum, use whole spectrum or partial
138 %   range with or without
139 %   baseline
140
141 ind1_ = ind1;
```

```
142 ind2_ = ind2;
143 if ~bCut % Use whole spectrum
144   ind1 = 1;ind2 = len_ch;
145 end
146 x1 = zeros(size(x));
147 x1(ind1:ind2) = x(ind1:ind2);
148
149 % t2 = [ind1_-5:ind1_ ind2_:ind2_+5];
150 % [t2,x2] = conv2col(t2,x(t2));
151 % % Linear fit of baseline with extended points
152 % [fBase,gof] = fit(t2,x2,'poly1');
153 % sigma_b = gof.rmse;
154 if bCut && bBase % Substract baseline from the spectrum
155   x1(ind1:ind2) = x1(ind1:ind2)-fBase(ind1:ind2);
156 end
157 % x1 = [ones(size(x1))*x1(ind1);x1;ones(size(x1))*x1(ind2)];
158 % Pad with horizontal lines on both sides
159 x1 = [zeros(size(x1));x1;zeros(size(x1))]; % Pad with zero on both sides
160 end
161
162 function p = check_peaks(peaks,s2s_fun,width,len_ch)
163 % Search for ranges without overlap
164
165 npeaks = size(peaks,1);
166 range = zeros(npeaks,2);
167 for i = 1:npeaks
168   mu = peaks(i,1);
169   smax = peaks(i,2);
170   sigma = s2s_fun(smax);
171   ind1 = max(floor(mu-width*sigma),1);
172   ind2 = min(ceil(mu+width*sigma),len_ch);
173   range(i,:) = [ind1 ind2];
174 end
175 i = 1;
176 N = 0;
177 p = zeros(size(range));
178 while i <= npeaks
179   N = N+1;
180   p(N,:) = range(i,:);
181   j = i;
182   while j < npeaks
183     if range(j,2) > range(j+1,1)
184       j = j+1;
```

```
185       p(N,2) = range(j,2);
186     else
187       break;
188     end
189   end
190   i = j+1;
191 end
192 p(N+1:end,:) = [];
193 end % EOF
194
195 function [centroid,area,area_unc,err] = weight(x,k,ind1,ind2,res,unc,unc_ratio)
196 % Get centroid and area using area uncertainty; peaks with difference of
197 %   channel number smaller than 'res' are considered as one peak
198
199 err = 0;
200 k = conv2col(k);
201 ch = find(k(ind1:ind2)~=0)-1+ind1;
202 i = 0;
203 N = 1;
204 len_k1 = length(ch);
205 peaks = cell(len_k1,1);
206 while i < len_k1
207   i = i+1;
208   peaks{N} = [peaks{N},ch(i)];
209   if i < len_k1 && ch(i+1)-ch(i) > res
210     N = N+1;
211   end
212 end
213 peaks(N+1:end) = [];
214 centroid = zeros(N,1);
215 area = zeros(N,1);
216 area_unc = zeros(N,1);
217 for i = 1:N
218   [centroid(i),area(i),area_unc(i)] = weight_ch(k,peaks{i},unc);
219 end
220 fprintf('\tCentroid     Area            Uncertainty         Ratio\n');
221 for i = 1:length(centroid)
222   fprintf('\t%-20.2f%-20.2f%-20.2f%-20.4f\n',centroid(i),area(i),...
223           area_unc(i),area_unc(i)/area(i));
224 end
225 for i = 1:N
226   if area_unc(i)/area(i) >= unc_ratio || sum(x(peaks{i})) < 1
227     area(i) = -1;
```

```
  end
end
centroid = centroid(area>0);
area_unc = area_unc(area>0);
area = area(area>0);
if length(centroid) == 1 && (centroid < ind1+5 || centroid > ind2-5)
  err = 1;
end
end % EOF

function [centroid,area,area_unc] = weight_ch(k,ch,unc)
% Get centroid from close channels with area weighted

[k,ch] = conv2col(k,ch);
area = sum(k(ch));
area_unc = sum(unc(ch))/length(ch);
centroid = sum(k(ch).*ch/area);
end % EOF
```

get_W.m

```
function [W,error] = get_W(x,scale,wname,bCheck)
% Get wavelet transform matrix
% If nargin > 3, check the error.
% 10/09/14

if nargin < 4
  bCheck = 0;
end

len_x = length(x);
x     = reshape(x,len_x,1);

stepSIG = 1;
precis = 10;
[val_WAV,xWAV] = intwave(wname,precis);
stepWAV = xWAV(2)-xWAV(1);
xWAV = xWAV-xWAV(1);
xMaxWAV = xWAV(end);

W = zeros(len_x);
```

```
21 a = scale;
22 a_SIG = scale/stepSIG;
23 j = 1+floor((0:a_SIG*xMaxWAV)/(a_SIG*stepWAV));
24 len_f = length(val_WAV(j));
25 f = [zeros(1,len_x-1),val_WAV(j),zeros(1,len_x-1)];
26 for i1 = 1:len_x+len_f-1
27   ind1 = len_f+len_x-1-i1+1;
28   ind2 = len_f+2*(len_x-1)-i1+1;
29   W(i1,:) = f(ind1:ind2);
30 end
31 len = len_x+len_f-2;
32 d = (len-len_x)/2;
33 first = 1+floor(d);
34 last = len-ceil(d);
35 W = diff(W);
36 W = -sqrt(a)*W(first:last,:);
37 error = 0;
38
39 if bCheck
40   buf1 = W*x;
41   buf2 = reshape(cwt(x,scale,wname),len_x,1);
42   error = norm(buf1-buf2);
43   figure('Name','W check');clf;
44   plot(buf1,buf2,'k-.');
45   axis equal;axis tight;grid on;
46 end
47
48 end
```

REFERENCES

[1] G. F. Knoll, *Radiation Detection and Measurement, 4th edition.* New York: John Wiley & Sons. Inc, 2009.

[2] S. M. Kay and S. L. M. Jr., "Spectrum analysis - a modern perspective," *Proceedings of the IEEE*, vol. 69, no. 11, pp. 1380–1419, 1981.

[3] G. R. Gilmore, *Practical Gamma-Ray Spectrometry, 2nd Edition.* Chichester, England: John Wiley & Sons. Ltd, 2008.

[4] A. N. Berlizov, "Limits of detection in the presence of the correlated back ground," *Journal of Radioanalytical and Nuclear Chemistry*, 2007.

[5] M. K. Cowles, *Applied Bayesian Statistics 1st Endition.* Springer, 2013.

[6] J. Stinnett, "Bayesian algorithms for automated isotope identification," M.S. thesis, University of Illinois at Urbana-Champaign, Urbana, 2014.

[7] M. B. Chadwick, P. Oblozinsky, and et al., "Endf/b-vii.0: Next generation evaluated nuclear data library for nuclear science and technology," *Nuclear Data Sheets*, vol. 107, pp. 2931–3060, 2006.

[8] M. A. Mariscotti, "A method for automatic identification of peaks in the presence of background and its application to spectrum analysis," *Nuclear Instruments and Methods*, vol. 50, no. 02, pp. 309–320, 1967.

[9] C. J. Sullivan, M. E. Martinez, and S. Garner, "Wavelet analysis of sodium iodide spectra," *IEEE Transactions on Nuclear Science*, vol. 53, no. 05, pp. 2916–2922, 2006.

[10] C. J. Sullivan, S. E. Garner, K. B. Blagoev, and D. L. Weiss, "Generation of customized wavelets for the analysis of gamma-ray spectra," *Nuclear Instruments and Methods in Physics Research, Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 579, no. 01, pp. 275–278, 2007.

[11] C. J. Sullivan and J. Lu, "Automated photopeak detection and analysis in low resolution gamma-ray spectra for isotope identification," *IEEE Nuclear Science Symposium Conference Record*, 2013.

[12] J. Lu, "Photopeak detection and quantification using wavelet analysis," M.S. thesis, University of Illinois at Urbana-Champaign, Urbana, 2013.

[13] X. Kong, "Advanced adaptive library for gamma-ray spectrometers," M.S. thesis, University of Illinois at Urbana-Champaign, Urbana, 2014.

[14] T. Burr and M. Hamada, "Radio-isotope identification algorithms for spectra," *Algorithms*, vol. 02, no. 01, pp. 339–360, 2009.

[15] M. Khider and B. Haddad, "An efficient 2D-WTMM and PNN approach to remove spurious radar echoes," *4th International Conference on Graphic and Image Processing*, vol. 8768, 2013.

[16] G. Strang and T. Nguyen, *Wavelets and Filter Banks.* Wellesley, MA: Wellesley-Cambridge Press, 1997.

[17] S. G. Mallat, *A Wavelet Tour of Signal Processing, 2nd Edition.* San Diego: Academic Press, 1999.

[18] D. B. Percival and A. T. Walden, *Wavelet Methods for Time Series Analysis.* Cambridge: Cambridge Univ. Press, 2000.

[19] C. L. Lawson and R. J. Hanson, *Solving least squares problems.* Englewood Cliffs, N.J.: Prentice-Hall, 1974.

[20] K. Faber and B. R. Kowalski, "Propagation of measurement errors for the validation of predictions obtained by principal component regression and partial least squares," *Journal of Chemometrics*, vol. 11, no. 03, pp. 181–238, 1997.

[21] M. Slawski, R. Hussong, A. Tholey, T. Jakoby, B. Gregorius, A. Hildebrandt, and M. Hein, "Isotope pattern deconvolution for peptide mass spectrometry by non-negative least squares/least absolute deviation template matching," *BMC Bioinformatics*, vol. 13, no. 01, 2012.

[22] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Communications on Pure and Applied Mathematics*, vol. 41, no. 07, pp. 909–996, 1988.

[23] H. Scheff, *The analysis of variance.* New York: Wiley, 1959.

[24] J. Stutz and U. Platt, "Numerical analysis and estimation of the statistical error of differential optical absorption spectroscopy measurements with least squares methods," *Applied Optics*, vol. 35, no. 30, pp. 6041–6053, 1996.

[25] K. Faber and B. R. Kowalski, "Propagation of measurement errors for the validation of predictions obtained by principal component regression and partial least squares," *Journal of Chemometrics*, vol. 11, no. 03, pp. 181–238, 1997.

[26] P. R. Bevington and D. K. Robinson, *Data reduction and error analysis for the physical sciences*. Boston: McGraw-Hill, 2003.

[27] R. Penrose, "A generalized inverse for matrices," *Proceedings of the Cambridge Philosophical Society*, vol. 51, pp. 406–413, 1955.

[28] M. James, "The generalised inverse," *Mathematical Gazette*, vol. 62, pp. 109–114, 1978.

[29] O. Alter, P. O. Brown, and D. Botstein, "Singular value decomposition for genome-wide expression data processing and modeling," *PNAS*, vol. 97, no. 18, pp. 10 101–10 106, 2000.

[30] M. G. Shirangi, "History matching production data and uncertainty assessment with an efficient tsvd parameterization algorithm," *Journal of Petroleum Science and Engineering*, vol. 113, pp. 54–71, 2014.